PH.D. THESIS

Optimal Control of Large Space Structures

by M.E. Baraka
Advisor: J.S. Baras

CSHCN Ph.D. 92-1
(ISR Ph.D. 92-3)

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **1992**	2. REPORT TYPE	3. DATES COVERED **-**

4. TITLE AND SUBTITLE
Optimal Control of Large Space Structures

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Office of Scientific Research,875 North Randolph Street,Arlington,VA,22203-1768

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR’S ACRONYM(S)

11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES **211**	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**			

OPTIMAL CONTROL OF LARGE SPACE STRUCTURE

by

Mohamed El Baraka

Dissertation submitted to the Faculty of the Graduate School
of The University of Maryland in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
1991

OPTIMAL CONTROL OF LARGE SPACE STRUCTURES

by

Mohamed El Baraka, Doctor of Philosophy, 1991

Dissertation submitted to the Faculty of the Graduate School
of The University of Maryland in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
1991

Advisory Committee:

Professor John S. Baras, Chairman/Advisor
Associate Professor Eyad H. Abed
Professor Carlos Berenstein
Assistant Professor Wijesuriya Dayawansa
Associate Professor Evaggelos Geraniotis

ABSTRACT

Title of Dissertation: **OPTIMAL CONTROL OF LARGE SPACE STRUCTURES**

Mohamed El Baraka, Doctor of Philosophy, 1991

Dissertation directed by: Dr. John S. Baras, Professor
Electrical Engineering Department
Martin Marietta Chair -
Systems Research Center

We present a computational spectral factorization method to solve the optimal state feedback control problem for flexible structures with the following features:

(1) Mathematically rigorous

(2) Wide range of applicability

(3) Flexibility of design

(4) Fast and Efficient

(5) Mini-computer (versus Super-computer) implementation. We apply this method to the following systems:

(1) A membrane

(2) A string

(3) An Euler-Bernoulli/Timoshenko beam models

(4) A Beam with structural damping and boundary control.

ACKNOWLEDGEMENT

I would like to express my gratitude and appreciation to Professor John S. Baras for his support during the preparation of this dissertation.

I also want to express my thanks to Mrs. Denise M. Best for her excellent typing of the manuscript.

Finally, I would like to acknowledge the partial support of the Systems Research Center through Air Force Office of Scientific Research AFOSR-90-0105A.

TABLE OF CONTENTS

Section Page

TABLE OF CONTENTS (Cont'd)

TABLE OF CONTENTS (Cont'd)

LIST OF FIGURES

LIST OF FIGURES
(Continued)

Figures Page

LIST OF SYMBOLS

(ARE)	Algebraic Riccati Equation
B.C.	Boundary Conditions
BVP	Boundary Value Problem
DPS	Distributed Parameter Systems
FFT	Fast Fourier Transform
I.C.	Initial Conditions
LQG	Linear Quadratic Gaussian
LSS	Large space Structures
ODE	Ordinary Differential Equations
PDE	Partial Differential Equations
iff	if and only if
$\forall$	for every
$\exists$	there exists

CHAPTER 1

INTRODUCTION AND REVIEW

Large space structures present new and challenging problems. The flexibility of these systems can not be ignored and distributed models (i.e. Partial Differential Equations) have to be devised.

We present a computational methodology to design optimal linear state feedback controllers that can be successfully implemented for linear systems governed by Partial Differential Equations. Efficient algorithms for infinite dimensional systems are hard to find as they all suffer from convergence problems and lead to poor performance in the high frequency range (spillover). First we consider the optimal shape control of a square flexible wire mesh using distributed electrostatic forces to minimize a quadratic criterion. This minimum norm problem is solved using the Projection method (i.e. Wiener-Hopf techniques). We use the Hilbert transform to express the projection of an L_2 function onto L_2^+, the subspace of functions with anti-causal inverse Fourier Transforms. This gives a closed form expression of the boundary values of analytic functions and leads to the spectral factorization theorem for scalar functions.

One of the main advantages of our approach is that the dimension of the problem is that of the control space and not that of the state space as in the Riccatti equation approach; which leads to a substantial reduction of computations. We also consider different dissipation mechanisms. The presence of damping (in physical systems) leads to a more difficult mathematical analysis, but on the other hand insures that the problems at hand are well defined (for instance computation of integrals by residues is valid only if no singularities lie on the boundary). Another desirable feature is the flexibility allowed the designer, since there are no constraints (in our method) such as optimal location of sensors/actuators (concatenation of sensors/actuators). Furthermore, the method is robust with respect to:

- unmodelled dynamics (nonlinear effects—)
- Variation in parameters —

and can be generalized to nonlinear systems and stochastic systems as well.

We study:

- the membrane problem
- the string problem
- beam problems (Euller-Bernoulli/Timoshenko)

We discuss briefly controllability and give a physical interpretation for wave problems although we do not go into the details.

Finally we analyze the beam with structural damping and boundary control where we achieve arbitrarily high performance with just one controller at the end, where robustness with respect to unmodelled dynamics is tested with success.

In the next few sections we would like to review some of the work that has been done in the area of control design for infinite dimensional systems. We compare it to what we did, and give a more precise discussion of our contributions.

1.1 Control of Flexible Spacecraft

In their paper on "Mathematical Modeling of Spinning Elastic Bodies" [46]. P.W. Likins, F.J. Barbera, and V. Baddeley study a distributed mass-finite element model of an elastic appendage attached to a rigid body. The model consists of an arbitrary number of elastic elements. Each contact point between two neighboring elements is called a node. The number of degrees of freedom, $6n$, of the system is determined by the number of nodes, n.

They state: "Such model, while a linear constant-coefficient ordinary differential system... seems to be the most promising model for rotating elastic structures."

"Algebraic complexities of a distributed-mass finite element model are so great that one obtains little useful insight into system behavior from these equations" [46].

We can also add that:

(1) The problem becomes even more intractable when you consider the control of this huge system ($6n \times 6n$).

(2) Even if you could get past all these difficulties and solve the control problem for this finite dimensional ($6n \times 6n$) system, in general it will be a poor approximation of the original problem (spillover, instabilities, high frequency behavior, etc.).

(3) In our method, since it is an eigenfunction expansion (therefore physical interpretation as a natural mode of the system), there is a considerable reduction in the dimension of the system.

(In reference [5, 6, and references therein] it is stated that a huge structure (such as SCOLE) while having hundreds of beam elements has only a few natural modes to be controlled).

In their paper on "Controllability and Observability for Flexible Spacecraft" [36], Peter E. Hughes and R.E. Skelton state the following:

"Assuming this minimal number of devices is present, the theorems indicate that for controllability or observability to be absent would be most exceptional. One would have to place force actuators exactly at nodes, for example, or have their direction exactly orthogonal to the model deflection at that point, in order to nullify the required condition."

In the same paper it is stated "the presence of small structural damping does not materially change the conclusions [in this paper] .... This qualitative change is not of practical significance ..."

(1) While "agreeing" on their first statement about Controllability/Observability, we add that our method allows the designer to have complete freedom to

locate the actuators and sensors, so as not to loose Controllability/Observability, but also allow ease of practical implementation.

(2) The importance of damping was overlooked in the literature for the following reasons:

(a) Its inclusion makes the mathematical analysis very involved (e.g. loss of closed form formulae).

(b) Its presence does not seem to have any impact on the Controllability/Observability properties of the system.

In our work we do include damping (viscous damping for the membrane, string, Euler-Bernoulli, and structural damping for beam with boundary control). The existence of damping will make the problems well-posed

(3) Let us also alert the reader to the following: Certain aspects of Controllability/Observability can only be understood in the context of PDE's. For instance: no matter how large the dimension N of the approximate system, it will be instantaneously controllable (which is not true for the original wave problem: waves take time to propagate).

Lattice structures offer additional complications. For instance in [51] we find the statement:

"Because of increasing interest in large lattice type structures for space applications, approximation of repetitive lattice grids with equivalent continuum models has gained popularity in recent years ... Conventional finite-element analyses of such grids with a very large number of nodes are expensive and time consuming ...".

We agree with their conclusion. This is especially true because the overall structure can have only a limited number ($n \leq 3$) of natural modes to be controlled, while the structure itself is made of hundreds (probably thousands) of elements.

One of the major advantages of our method is its ability to control huge discrete space structures by controlling an equivalent continuum model.

We can treat all one dimensional structures (such as strings, Euler-Bernoulli Timoshenko beams ...) exactly by our scalar spectral factoring theorem; see Theorem 2.13 in section 2.3.4.

In his paper "Trends in Large Space Structures" [6] Mark J. Balas states:

"Conditions on (A, B) for existence/uniqueness of gain operators G^* have been given, but these conditions are quite stringent; in addition, solving the nonlinear PDE is not easy. Finally it may turn out that the on-board controller cannot implement the optimal control law. Therefore some approximations and simplifications of the DPS control problem will be required to produce a suboptimal but implementable controller ..."

However in our approach we give a powerful algorithm for computing the optimal control without any simplification or approximation to the original control problem. Moreover, for the practical implementation, great flexibility is left to the designer to choose the configuration of the system.

We also draw attention to two difficulties associated with Riccati differential systems:

(1) Any algorithm must be able to handle very large scale nonlinear systems.

(2) They also turn out to be stiff problems.

Reduced order models try to alleviate the above difficulty but they make spillover worse (i.e., unstable design).

It has also been "shown" and has been observed in experimental systems that few actuators/sensors are needed for a space structure to achieve controllability and observability [6]. For instance all one dimensional structures need only one actuator/sensor, which can be handled exactly by our algorithm; see Theorem 2.13 in section 2.3.4. The reason, however, people try to put as many sensors/actuators is to alleviate spillover (not to increase Controllability/Observability) which has been shown to fail to stabilize the system.

Any controller based on a finite dimensional approximation of a (distributed parameter system) will suffer from interactions with three kinds of residuals:

- modelling error term (and its dual)
- control spillover
- observation spillover

These interaction terms cause instabilities. Such instabilities have been demonstrated in flexible structures laboratory experiments at C.S. Draper Lab and Jet Propulsion Lab, as well as in numerical simulation ([6] and references therein)

In the literature people suggest the following:

In the very special case where a finite dimensional approximate model is used and the actuators (or sensors) can be located so that the control (or observation) spillover term is zero one can limit spillover influence but then you lose Controllability/Observability. In the same paper [6]. Balas states:

"The dimension of the system may be too large for successful on-line implementation".

"There are no fully developed tools (for spillover compensation)".

"There are no easy solutions or panaceas".

Point devices could be located at or near as many zeros of residual modes as possible, however this leads to loss of observability/controllability and large controller gains (therefore nonlinear controller behaviors) add the practical difficulty of physical location for devices on actual structures. Balas continues on to say, "The most promising method is the use of filtering to remove the residual modes ... filters however introduce signal distortion and delay and while they may remove some instabilities generated by spillover they will introduce other instabilities due to filter delay. Also in the same reference it is stated:

“Locating actuators and sensors at (or very near) the zeros of the mode shapes will make it impossible to retain Controllability/Observability. Also dependence on exact locations can lead to highly sensitive designs”

In noisy systems these suggested remedies become even more doubtful.

Balas in his paper on“Model Control of a Simply Supported Beam” [7] has the following set up: An Euler-Bernoulli beam is controlled by a single point actuator (at 1/6 length of the beam) and a single point sensor (at 5/6) and the feedback controller controls the first three modes of the beam. When given an initial displacement so that the first three modes of the beam have an initial displacement of one unit, the decay in the three controlled modes is equivalent to 1% damping (they are trying to have small gains not to destabilize the system). The main problem however is that they observe residual mode instability caused by spillover.

We also mention direct Velocity Feedback [6] (termed active damping). It does not lead to spillover instabilities, however it suffers from the following restrictions:

- an equal number of collocated force actuators and velocity sensors.
- the actuators must not excite zero frequency modes.

Moreover the conclusions arrived at need to be taken cautiously as the analysis assumes a distributed control, while any practical implementation will certainly have a finite number of actuators/sensors and this fact often leads to unstable designs (see [24]), and this has been observed by structural engineers.

We need to stress the fact that high gains cannot be handled by usual numerical methods as already borderline unstable algorithms will certainly go unstable when one uses a high gain controller. This will also lead to nonlinear behavior of the controller which can be handled only by a robust algorithm. However high gains is what is needed for accurate pointing or attitude control.

1.2 Electrostatic Control of Electromagnetic Reflectors

The use of distributed electrostatic forces to control the shape of a flexible membrane has allowed the construction (of a wide variety) of precise electromagnetic reflectors [42, 43, 44] with fractional-meter to kilometer aperture diameters and operate over the centimeter to optical wavelength range, for ground-based and space based operation. However large curvature or diameter leads to a reflector equilibrium shape exhibiting Rayleigh- Taylor deflection instabilities [42, 43, 44]. Experimental stabilization studies on systems governed by hyperbolic partial differential equations have been reported:

(1) a spring in tension which was excited then stabilized electrostatically using 2 sensors/actuators).

(2) fluids which were excited and then stabilized.

(3) A compressed beam which was excited mechanically and then stabilized magnetostatically with 1 sensor/actuator.

None were successful in stabilizing more than 1 mode.

Later Lang et al. [42] did experiments on a two dimensional wire-mesh continuum which was excited and then stabilized electrostatically. The performance criterion was such that when it approaches unity it was an indication of control system inability to maintain mesh stability due to insufficient actuator influence. An approximation to the model experimental system was used to design a near-optimal control system (with a limited number of primary modes $m, n \leq 4$).

"The linear LQG control system just derived proved to be unsatisfactory at high bias voltages due to physical reasons not modeled... state augmentation, to include the imbalance, proved unsatisfactory due to a resulting 2.5 fold decrease in feasible control system sampling rates." [42]

"... when stabilization failed, failure was nearly always explained by the destabilization of an unmodeled mode or from excessive membrane deflection driven beyond the actuator capture range by actuator or sensor noise. Finally, actuator and sensor spillover were experimentally found to be a predictable cause of closed loop system destabilization ...

"The second difficulty experienced with the LQG control systems was their online computational complexity necessitating slow sampling rates. The present experimental system was restricted to the LQG control of three modes due to sampling consideration alone".

This work of Lang et al. [42] was the primary justification for our developing this algorithm to solve "the membrane problem" (Chapter 2) and linear distributed parameter systems in general (String (Chapter 3), beam (Chapter 4), Boundary control (Chapter 5)). It is significant to point out that our algorithm allows the incorporation of as many as 400 modes ($m \leq 20, n \leq 20$) for the membrane (compared to 3 modes above), without any control or observation spillover and excellent robustness to unmodelled dynamics (which was tested for the more complicated problem of boundary control of a beam in Chapter 5). Lang's experiments have also shown what kind of control technology is necessary for very precise electrostatic flexible membrane reflectors (e.g. tolerance 10^{-5} times the reflector diameter see [43]).

They also state: "The design of controllers for distributed parameter systems from tuncated modal system descriptions is inherently conducted in ignorance of the actuator and sensor coupling to the unmodelled modes. The real presence of this coupling, referred to here as spillover, can cause significant changes in the expected closed-loop system behavior. To examine the effect of spillover, this coupling is treated as a perturbation" [45].

It is our view that this kind of perturbation analysis while giving insight into the spillover problem, does not eliminate it. This complicated perturbation analysis, shadows the original control problem, and does not lead itself into inclusion in an automated algorithm to solve the control problem for distributed parameter systems.

1.3 Spectral Factorization Versus Riccati Equation Approach

Various algorithms for spectral factor calculation have been proposed. Some are based on algebraic methods and rely on rational approximations of the transfer function and large scale linear systems. As such they are not suitable for distributed parameter systems.

J.H. Davis and R.G. Dickinson in [23] propose to seek a recursive algorithm for spectral factorization inspired by the Newton-Raphson algorithm for the solution of the algebraic Riccati equation. To describe their algorithm define:

$F(jw) = I + S(j\omega)$, the function to be spectral factored and let P_+ be the causal projection operator on the convolution algebra $I \oplus L_1$ (or L_2) defined by:

$$P_+[I + \int_{-\infty}^{+\infty} F(t)e^{-j\omega t}dt] = I + \int_{0}^{\infty} F(t)e^{-j\omega t}dt \tag{1.1}$$

Then the Davis-Dickinson iteration is given by:

$$F_{n+1} = P_+[(F_n^*)^{-1}H(F_n)^{-1}]F_n \tag{1.2}$$

Independently of any Riccati equation considerations they prove under some mild conditions (satisfied for distributed systems of interest which include the membrane, string, beams, etc.) and with a suitable choice of the initial condition, the iterate converges to the desired spectral factor; and this is a major achievement.

We would like, however, to stress the following two facts:

- Of utmost importance is the suitable choice for F_0, the initial iterate for the factor. As suggested by Davis-Dickinson a suitable choice that insures convergence is to take for F_0 the diagonal matrix of the spectral factors of the diagonal elements of $F(j\omega)$. Therefore one crucial component of the algorithm is how to "achieve" scalar spectral factorization. Our contribution was to give a closed form formula based on the Hilbert Transform.

- The other crucial element in the Davis-Dickinson iteration is the projection P_+ based on Stenger's algorithm [64]. Our contribution is to give a closed form

  formula based on the Hilbert Transform.

- With these two major modifications we get a superior algorithm with the following features:

(1) requires few points compared with the original Davis-Dickinson iteration scheme. For instance:

their algorithm applied to a "relatively" smooth 3×1 transfer function given by a closed form formulae needs about 4096 points to give a 10^{-1} "trace" difference. (See [23]). By contrast our algorithm applied to the 4×4 transfer function of a membrane with singularities (more than 8 singularities smoothed somehow by damping, see Figure [16]) needs 20 points to give a precision higher than 10^{-7}.

Note: Our contention is that for a complicated real life system such as the membrane one would actually need many more points than 4096 points, i.e. you would have to solve tens (or hundreds) of thousands of systems of complicated partial differential equations. (See Chapter 2 for example to get an idea about the complexity of such systems).

(2) Our algorithm gives the actual maximum error on every entry of the spectral factor matrix, instead of just a trace measure of the error.

(3) Our algorithm converges very fast: For a comparison:

- Their algorithm needs 7 iterations (and 1024 points) to achieve a 10^{-4} trace tolerance for a rational 2×2 transfer function matrix.

- In comparison ours converges in 1 iteration (and $20 points$) to achieve a precision higher than 10^{-4} for every entry of the 4×4 transfer function of the membrane (irrational, not given by closed form formula ...).

(4) Our algorithm is completely automated:

(a) One of the main features of our program is its use of adaptive integration [57].

(b) As input it needs:

- the frequency range
- the number of points in frequency range
- the absolute and relative errors allowed
- the number of iterations
- some machine dependent constants (with programs provided to compute them).
- maximum number of function evaluations allowed (500 was enough for all our programs)

It then gives the output with the desired precision (the number of function evaluations if so desired) and possibly some flags alerting to possible troubles:

- If more than the number of function evaluations allowed is necessary, the algorithm asks whether to use more iterations and continue to achieve the desired precision.

- If unexpected singularities occur you will be alerted to their location.

For the membrane problem we asked for a precision of 10^{-1} (with 20 points in the frequency range) the algorithm returns with a precision higher than 10^{-4} in the first iteration. We stress again the fact that it is an automated algorithm. All it needs is the values of the transfer function (it will tell you if you need more points to achieve the desired precision). To state that it is an automated algorithm it means that "no analysis" of the problem is necessary; the algorithm does it all. In contrast:

(c) The Davis-Dickinson algorithm for instance needs a thorough preliminary analysis of the problem before it can be applied. For instance in [23] they say "previous experience with the kind of functions involved in H suggested a close analysis is needed." They study a 3×1 irrational transfer function (given in a closed form), they had to determine where the peaks occur (for every entry) and their width then use different grid scales accordingly. They go on to state "it is clear from table 1 that the location of the peak played a key role in choosing the grids". The fine grid for instance was chosen so that more points (10 in their example) were chosen near the peak.

They also separate the finite dimension, the "delay line" and the part containing the peaks and do spectral factorization of these factors (with different grid scales accordingly) to later recover the spectral factor of the whole transfer function H. Even then (with as many grid points as 4096 in the fine grid) only a trace tolerance of 10^{-1} was achieved in 3 iterations.

(d) This shows why it will not be possible to use such an algorithm when we have a complicated system. In the case of the membrane for instance (Chapter 2) we have a 4×4 matrix with more than 8 singularities in each term. It would be close to impossible to do the kind of close analysis required: How to get all the information when the transfer function is nearly never given in a closed form, but only as a solution to a system of PDE's (at each point).

(5) Let us try to explain one of the major difficulties present in the original Davis-Dickinson algorithm. The algorithm relies on the Stenger's formula for the projection and uses the FFT to compute the required transforms and convolution. This is to say that the algorithm is in a fundamental way a 1^{st} order approximation and that is the source of the problem. The FFT is just a fast way of computing the Discretized

Fourier Transfer which is a first order approximation. Therefore all the well-known difficulties associated with the discretized Fourier Transform come to play here. Let us remember that the transfer function for distributed systems will contain singularities (see the problems we treated in the dissertation) and this is why one needs a huge number of points (because of uniform sampling) to achieve a limited tolerance when all we need is to concentrate the grid points near the singularities otherwise the sampling rate will be that near the worst singularity.

(6) We would like to mention one interesting feature of our scalar spectral factorization formula. It allows the computation of the spectral factor at a desired point. For the other iterative algorithms it is all or nothing, in other words you have to compute the spectral factor at all points. This makes them unsuitable for automatic integration, because an adaptive integrator will try to achieve the desired precision with the given number of points. If it is unable to achieve the desired precision, it will ask for more points.

(7) These comments and comparisons (while are substantiated in the main body of the thesis) indicate that we have developed superior spectral factorization algorithm to solve the optimal regulator problem (and the filtering problem for distributed parameter systems as well as for finite dimensional systems). We also intend to exploit the connections of spectral factorization to a host of applied mathematics and engineering problems to provide for the first time a powerful computational tool. P.B. Molinari has investigated "Equivalence relations for the algebraic Riccati equation" [53] where he showed that "stabilizing" solutions of the finite dimensional (ARE) are in one to one correspondence with certain "spectral factorizations" [40] of a real rational matrix. Also, T. Kailath in his paper on "Fredholm Resolvent, Wiener Hopf-equation and Riccati Differential Equations" studies the relationship between

these mentioned problems (even though his thrust was to reduce the spectral factorization to a Riccati equation). He does however state: "despite its wide use in modern control theory there is no universally accepted "best" way of solving Riccati equations ...". Jan C. Willems in his paper "Least Squares Stationary Optimal Control and the Algebraic Riccati Equation" [69] derived time-domain and frequency domain conditions for the existence of solutions which are then classified.

It is already apparent in his classification of solutions that the problem is very complicated. For instance: every real symmetric solution of (ARE) will be given by: $K = K^+P + K^-(I - P)$ (where P is a projection (and you need to solve eigenvalue problems)

$$ex : K^2 = I \Rightarrow K^+ = I, K^- = -I \text{ and } K = -I + 2P \tag{1.3}$$

$$P = \begin{bmatrix} \cos^2\sigma & \sin\sigma \cdot \cos\sigma \\ \cos\sigma \cdot \sin\sigma & \sin^2 \cdot \sigma \end{bmatrix} \tag{1.4}$$

He also gives an eigenvalue condition on the initial iterate to make the Newton-Raphson algorithm converge. The linear quadratic regulator is one of the best studied problems in Control theory, yet many aspects of the problem are still unsolved; for instance the sensitivity of the solution to (ARE) under perturbations and the compuational performance of the algorithms have not been examined in sufficient extent.

The situation for infinite dimensional systems is even worse. Let us give the example of the membrane (Chapter 2): to have convergence we needed to take 400 ($m \leq 20, n \leq 20$) terms in the eigenfunction expansion. This leads to a state space dimension $N = 400$ and therefore we need to solve a 400 × 400 Riccati equation (which would be a poor approximation anyway because of spillover). By contrast, the Wiener-Hopf approach using our computational algorithm requires just one closed form exact scalar spectral factorization.

J.H. Davis and M. Barry in their paper on "A Distributed Model for Stress Control in Multiple Locomotive Trains" [20] show that the frequency domain calculation was found computationally efficient (even with the shortcomings of their alogirthm as we explained earlier) while a Riccati equation truncation proved numerically difficult to handle.

There is also a huge literature on infinite dimensional Riccati equations and we refer to [6] and the references therein.

Next we would like to point out a different direction for designing control systems. Design methods based on frequency domain techniques have also been extended to distributed systems. They use properties of analytic function matrices (Coprime factorization - Bezout identity ...) in appropriate Banach algebras (See Baras [10], Baras et al. [11, 12], Callier and Desoer [15]). We are unaware of algorithms to solve their equations (Bezout identity for instance).

R.F. Curtain [19] studies the robust H^∞ desing for some classes of infinite dimensional systems. For such designs there is a trade-off between:

- the robustness margin
- the order of the controller
- the reduced order model.

These may lead to poor performance, in addition to computational difficulties. Furthermore, there are no efficient algorithms except for reducing the H^∞ design problem to a Riccati equation [23].

1.4 Numerical Simulation of Hyperbolic PDE's

Galerkin methods (and in particular finite element methods) have had a great success in solving elliptic (and parabolic) equations which made them very popular in the engineering community. In contrast, these methods lead to poor performance when applied to hyperbolic problems.

Recently, they were modified to handle first order hyperbolic systems; the so called discontinuous Galerkin [37, 38] methods but are not convenient to incorporate as part of a control design system. For second order hyperbolic systems the problem is still open.

In our work (see section 2.1.2) we implemented the method of lines which leads to large scale systems of ODE's, and turned out to be unstable. This led us to choose eigenfunction expansions as a tool of solving hyperbolic systems of second order and we implemented it for the following problems:

- the membrane
- the string
- the Euler-Bernoulli beam
- the Timoshenko beam
- the beam with boundary control

The numerical results are excellent. To compute the eigenfunctions when no closed form exists we suggest a finite element approach (see C. Weeks [70]) as we mentioned earlier very few need to be computed which makes the method very efficient. Another major advantage is that eigenfunctions correspond to the natural modes of the physical system and fit perfectly in a control algorithm . Let us mention that eigenfunctions have been used as a theoretical tool to investigate controllability (D. Russel [59], [60]). C. Weeks used them for static shape determination and control for a large space structure. J.H. Davis used eigenfunction expansions to compute the filter gains for a train model [21]. Let us mention also the work by H.T. Banks et al. [8, 9] who used finite elements for parameter estimation for distributed system and implemented their algorithm on vector computers (Cray 1-S).

In the algorithm developed in this thesis, eigenfunction expansions are incoprorated as part of an optimal control algorithm for infinite dimensional systems.

1.5 Description of the Algorithm

Referring to Figure 14, we see that there are four major blocks:

1 - "Hyperbolic system" block

2 - "Elliptic system" block

3 - "Spectral Factorization" block

4 - "Optimal Gains" block

5 - "Fourier Transform" block

- Block 1 is responsible for the simulation of the forward system and contains two main subblocks:

  a - "Eigenstructure block" which computes the eigenvalues and eigenfunctions of the system.

  b - "Initial Value problem" block which computes the time varying coefficients in the eigenfunction expansion.

  This block gives the "full state" of the system; for example it will compute the displacement and the velocity (no differentations involved).

- Block 2 is responsible for the simulation of the system in the frequency domain and is on a theoretical level equivalent to Block 1; in other words we can recover one from the other in principle. It has two main subblocks:

  a - The "Green's Function block"

  b - The subblock computing the solution to the elliptic system.

- Block 3 is responsible for the spectral factorization of the "modified" transfer function and has two main subblocks:

  a - The scalar spectral factorization algorithm based on the Hilbert transform.

  b - Newton-Raphson iteration for the full matrix case [23]

- Block 4 computes the Optimal gains using the J. Davis formulae [22].
- Block 5 allows the passing from the frequency domain to the time domain.

This is a brief description of the complete algorithm, but one cannot have a full understanding unless one studies an example, therefore we suggest reading Chapter 2 on the membrane because it was the first problem that led us to this algorithm and therefore contains the full details. It would also be convenient to skip section 2.3 on spectral factorization (thinking of Block 3 as a blackbox) without any break in the logic (understanding) of the Chapter. The next step would be to apply the algorithm to a "simple" one dimensional structure (and thus many interesting large space structures) to gain a full mastery.

CHAPTER 2

THE LINEAR REGULATOR PROBLEM OF A VIBRATING MEMBRANE: MATHEMATICAL AND COMPUTATIONAL FRAMEWORK

In this chapter we consider the optimal shape control of a square flexible wire mesh using distributed electrostatic forces so as to minimize a quadratic criterion.

By using an appropriate Hilbert space set-up, it is possible to show that the optimal control is the solution to a minimum norm problem that can be solved using the projection method (i.e. Wiener Hopf technique here).

In section 2.1 we review the hyperbolic system associated with the vibrating membrane. We show existence and uniqueness of solutions using the semi-group approach in an appropriate cartesian product of Hilbert spaces. It is also shown that the corresponding infinitesimal generator satisfies the hypotheses of the Hille-Yosida-Phillips theorem and we deduce the exponential stability of the semi-group (see Phillips [34] and also Yosida [71, 72]).

Subsequently, we solve this system numerically using the method of lines which transforms the hyperbolic system into a set of linear differential equations. Although the convergence of this method has been proved (see Teman [65]) the numerical results are not satisfying and the method is time consuming. This leads us to seek an eigenfunction expansion of the solution which gives satisfactory numerical results by:

- avoiding numerical differentiation
- precision "independent" of the mesh size
- precision only limited by the capacity of the computer

In section 2.4 we apply the Fourier transform to the wave equation to get an elliptic 2^{nd} order equation with a complex parameter that we propose to solve by the Greens operator technique which is a special case of the Hilbert-Schmidt theory of integral operators. To this end we expand the "Real" Green's function into an exponentially uniformly converging series. To alleviate the amount of computations, in particular to avoid the implementation of double integrals numerically, we devise a first order discretization of the integrated Green's function. Next we extend the above results to the case of a complex parameter (physically presence of damping) by using the principle of analytic continuation for anlaytic functions. We then solve the elliptic system and present the numerical results.

In section 2.5 we use the Hilbert transform technique to express the projection of an L^2 function onto L_2^+, the subspace of functions with anti-causal inverse Fourier transform. Again we use the Hilbert transform to express the boundary values of analytic functions. Combining these results we get the spectral factorization theorem for scalar functions. The numerical results obtained are also reported.

In the last section 2.4.4 we present to give the gain computations and also report the numerical results.

First we compare the Riccati equation approach to the Wiener-Hopf technique for a scalar 1-dimensional control problem. Even in this very simple example one gets some interesting insight: The estimation of the spectrum of the transfer function is crucial in the computation of the gains. In section 2.6 we compute the resolvent operator using the theory of the elliptic system developed in section 2.4.

In section 2.6 we give formulae for computing double Fourier series coefficients of the control functions and give numerical results.

Then we present the control problem of the membrane when the control is a scalar function located at a certain point of the membrane. We also describe the multidimensional version which is very complex and shows that the successful numerical implementation will depend on:

- choosing a good notation
- all blocks of the system are working perfectly

We then study the control of the membrane along a mode of the system which is a special case of the previous one and show how this assumption simplifies the formulae greatly.

Subsequently, we provide a comparison between different approaches of controlling the membrane and stress some "hidden" theoretical difficulties when one tries to implement these approaches practically. It becomes also apparent that when for practical computations (involving operators in infinite dimensional spaces) we must satisfy the following requirement.

1. Need to devise stable numerical algorithms
2. Make sure that one can achieve a high accuracy in a reasonable time for all building blocks.
3. Test the algorithms in the same condition as when all blocks are put together, especially because of the memory storage problem on a minicomputer.
4. Select easily implementable notations and apply structured programming principles (see multi control section 2.6 for illustrations).

As will be seen the numerical results obtained in this chapter are satisfactory. Our main contributions in this chapter are as follows:

- The Numerical Implementation of the Method of lines (section 2.1.2).
- The Numerical Implementation of the eigenfunction expansion for the "hyperbolic" system (section 2.1.3).
- The computation of the Green's function for the "membrane" (section 2.2.1).

- The Numerical Implementation of the solution to the elliptic system (section 2.2.7).
- The scalar spectral factorization theorem (section 2.3.3).
- Optimal Control of one dimensional structures theorem (section 2.3.4).
- Numerical Implementation of the scalar point control for the membrane (section 2.3.5).
- Numerical Implementation of the multidimensional control problem for the membrane (section 2.4.5).
- Numerical Implementation of the mode shape control problem (section 2.4.6).

2.1 Study of the Hyperbolic System

A large class of physical systems, such as:

- distributed systems described by linear partial differential equations
- delay differential systems
- lumped systems described by linear ordinary differential equations

can be given a unified treatment using semi-group theory. This is done by formulating the above problems as abstract differential equations in an appropriate Banach (or Hilbert) space:

$$\begin{cases} \dot{x}(t) = Ax(t) + u(t) & t\epsilon(0,\infty) \\ x(0) = x_0, \quad x_0 \epsilon D(A) \end{cases} \tag{2.1}$$

$x(t), u(t) \epsilon \mathcal{B}$ Banach space.

A is generally a closed linear operator with dense domain in B and is called the infinitesimal generator of the system.

If we restrict our attention to the unforced system:

$$\begin{cases} \dot{x} = Ax \\ x(0) = x_0 \end{cases} \tag{2.2}$$

and assume that the problem is well-posed then the solution $x(t)$ will define a semi-group of bounded operators on $\mathcal{B}$: $x(t) = T(t)x_0$. $\{T(t)\}t \in [0,\infty)$ is called the semi-group generated by A and satisfies:

(1) $T(t,s) = T(t)T(s)t, s \geq 0$ (semi-group property)

(2) $\lim_{t\downarrow 0} T(t)x_0 = x_0$ (C_0-property or strong continuity)

It is the norm defined on the Banach space $\mathcal{B}$ (most important ones are Sobolev spaces) that determines the degree of smoothness of the solution (strong versus weak solutions).

In Section 1 we shall attempt to put the wave equation governing the motion of the membrane into this general framework, and apply the general theory of semi-groups to get some interesting insight about the solution.

2.1.1 Semi-group Approach to the Wave Equation

The equation governing the motion of a vibrating membrane is given by:

$$\begin{cases} \dfrac{\partial^2 h}{\partial t^2} = a\dfrac{\partial^2 h}{\partial x^2} + b\dfrac{\partial^2 h}{\partial y^2} + c\dfrac{\partial h}{\partial t} + dh + u \quad \text{in } \overset{\circ}{\Omega} \\ h(x,y,0) = h_0(x,y) \\ h_t(x,y,0) = v_0(x,y) \qquad \text{Initial Conditions (I.C.)} \\ h(x,y,t)|_{\partial\Omega} = 0 \qquad \text{Boundary Conditions (B.C.)} \end{cases} \tag{2.3}$$

where Ω is given by:

$$\Omega = [0,\ell] \times [0,\hat{\ell}] \tag{2.4}$$

where h is the displacement of the point (x,y) on the membrane. c is the damping coefficient. The term dh comes from linearizing an originally nonlinear equation. u is the control applied to the membrance.

To investigate the existence and uniqueness of solutions to the above second order equation we transform it into a two-dimensional hyperbolic system of first order.

$$\begin{cases} \dfrac{d}{dt}\begin{bmatrix} h \\ h_t \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ a\frac{\partial^2}{\partial x^2} + b\frac{\partial^2}{\partial x^2} + d & c \end{bmatrix} \begin{bmatrix} h \\ h_t \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \\ I.C. \\ B.C. \end{cases} \tag{2.5}$$

We define the following partial differential operators:

$$L = -\left(a\frac{\partial^2}{\partial x^2} + b\frac{\partial^2}{\partial x^2} + d\right)$$

$$A = \begin{bmatrix} 0 & 1 \\ L & c \end{bmatrix} \tag{2.6}$$

with appropriate boundary conditions $(B.C.)$

(a) Sobolev Spaces:

We introduce some spaces that are going to be useful later. $H_k(\Omega)$(k integer ≥ 1) consists of all functions u having partial derivatives of order $< k$ in L^2 (Ω), where the derivatives are understood in the sense of distributions. $H_k(\Omega)$ is a Hilbert space when equipped with the inner product:

$$(u,v)_{H_k(\Omega)} = (u,v)_{L^2(\Omega)} + \sum_{j=1}^{k}(D^j u, D^j v)_{L^2(\Omega)} \tag{2.7}$$

where $D^j = \frac{\partial}{\partial xj}$ and $H_k(\Omega)$ is called the Sobolev space of order k. We also introduce the space $\mathcal{D}(\Omega)$ consisting of all C^∞ functions with support in Ω. $\overset{\circ}{H}_1$ (Ω) is the closure of $\mathcal{D}(\Omega)$ in $H_1(\Omega)$. For a thorough treatment of Sobolev spaces see the monograph by Adams [1].

Next we give some important properties of L.

We consider L acting on $L^2(\Omega)$ (in the sense of distributions), with domain $D(L) = H_2(\Omega) \cap \overset{\circ}{H}_1$ (Ω), equipped with the usual inner product: $(u,v)_{L^2(\Omega)} = \int\int_\Omega uv$

L is self-adjoint [74]:

To make the computations easier we first make the change of variables:

$$\begin{cases} \alpha & = \sqrt{a}x \\ \beta & = \sqrt{b}y \end{cases} \tag{2.8}$$

In these new coordinates L becomes:

$$Lu = -\Delta u - du \tag{2.9}$$

If $\mathcal{D}$ is the transformed domain then: $u|_{\partial\mathcal{D}} = 0$

By elementary computations we get:

$$\begin{aligned} \iint_{\mathcal{D}} u \cdot Lu &= \iint_{\mathcal{D}} \nabla \cdot (v\nabla u - u\nabla v) \\ &= \int_{\partial\mathcal{D}} (v\nabla u - u\nabla v) \cdot n \quad \text{(by the Divergence theorem)} \\ &= \int_{\partial\mathcal{D}} v\frac{\partial u}{\partial n} - u\frac{\partial v}{\partial n} \equiv 0 \quad \text{(because of the boundary} \\ &\qquad\qquad\qquad\qquad \text{conditions } u|_{\partial\mathcal{D}} = v|_{\partial\mathcal{D}} = 0\text{)} \end{aligned} \tag{2.10}$$

which leads to:

$$(u, Lv) = (v, Lu) \qquad \forall u, v\epsilon D(L) \tag{2.11}$$

Hence L is self-adjoint.

We can also show by elementary arguments that the eigenvalues of L are real; and that eigenfunctions corresponding to different eigenvalues are orthogonal:

$$(\varphi_i, \varphi_j) = 0 \quad (\text{for } \lambda_i \neq \lambda_j) \tag{2.12}$$

L is positive:

We use the completeness of the eigenfunctions of the self-adjoint operator L (see [31, 37]) to write:

$$\varphi = \sum \alpha_k \varphi_k \quad (L\varphi_n = \lambda_n \varphi_n) \tag{2.13}$$

Therefore $(L\varphi, \varphi) = (\sum \alpha_k \lambda_k \varphi_k, \sum \alpha_n \varphi_n) = \sum \alpha_k^2 \lambda_k$ (by orthogonality of eigenfunctions). In section [?] we will prove that the eigenvalues of L are given by:

$$\lambda_{mn} = a\left(\frac{n\pi}{\ell_x}\right)^2 + b\left(\frac{m\pi}{\ell_y}\right)^2 - d; \quad m, n = 1, 2, \cdots \tag{2.14}$$

We summarize the previous discussion:

Theorem 2.1:

The self-adjoint operator L defined in 2.6 is positive i.e. $(L\varphi, \varphi) \geq 0$ (and equals 0 only for $\varphi = 0$), iff the following condition holds:

$$\pi^2\left(\frac{a^2}{\ell_x^2} + \frac{b^2}{\ell_y^2}\right) - d > 0 \tag{2.15}$$

The domain $D(L)$ is L is dense and the following estimate holds:

$$(L\varphi, \varphi) \geq \left(\pi^2\left(\frac{a^2}{\ell_x^2} + \frac{b^2}{\ell_y^2}\right) - d\right) \cdot \| \varphi \|^2, \quad \varphi \in D(L) \tag{2.16}$$

Now we want to quote some key results from semi-group theory [18, 34, 71, 72] that will allow us to get the existence and uniqueness of solutions to the "membrane" equation. The first major theorem we quote is the Hille-Yosida-Phillips Theorem which characterizes generators of C_0 type (strongly continuous) semigroups.

Theorem 2.2 (Hille-Yosida-Phillips [18, 34, 71, 72]):

Let A be a closed linear operator with dense domain in a Banach space X. The A generates a C_0 semi-group iff:

$(i) \exists M, \omega_0; \forall \lambda > \omega_0 : \lambda \epsilon \rho(A)$ (resolvent set of$A$).

$$(ii) \|R(\lambda; A)^n\| \leq \frac{M}{(\lambda - \omega_0)^n} n = 1, 2, \ldots (R(\lambda; A) \text{ resolvent of } A) \tag{2.17}$$

Under these conditions we have:

$$\forall \omega > \omega_0, \exists M \text{ such that: } \forall t \geq 0, \|T_t\| \leq M e^{\omega t} \tag{2.18}$$

where T_t is the semi-group generated by A.

The second major theorem we quote is the Lumer-Phillips theorem which is a consequence of the previous theorem and gives a different characterization of semi-group generators.

Theorem 2.3 (Lumer-Phillips [18, 34, 72]):

Let A be a closed, densely defined, linear operator on a Banach space X. Then A generates a semi-group T_t on X satisfying:

$$\|T_t\| \leq e^{\omega t}, \ \forall t \geq 0 \tag{2.19}$$

iff

$$\begin{aligned} \forall \lambda > \omega : \|(\lambda I - A)x\| &\geq (\lambda - \omega)\|x\|, \ x \epsilon D(A) \\ \|(\lambda I - A)x^*\| &\geq (\lambda - \omega)\|x^*\|, \ x^* \epsilon D(A^*) \end{aligned} \tag{2.20}$$

where $D(A)$ and $D(A^*)$ designate the domains of A and A^* respectively.

Remarks:

1. Condition 2.20 will be satisfied if A is dissipative.
2. The condition 2.19 is not really a restriction as we can define a new equivalent norm:

$$|||x||| = \sup e^{-\omega t}\|T_t x\| \tag{2.21}$$

3. Actually we are going to specialize it to the case of a Hilbert Space (next corollary) which will be the version we are going to use to prove existence and uniqueness of solutions.

Corollary 2.3:

Let A be a closed, densely defined, linear operator or a Hilbert space H. Then A generates a semi-group T_t on H satisfying $\| T_t \| \leq e^{\omega t}$ for a suitable ω and $\forall t \geq 0 iff$

$$\begin{aligned} \exists k : Re\{(Af, f)\} &\leq k \parallel f \parallel^2, \quad \forall f \epsilon D(A) \\ Re\{(A^* f, f)\} &\leq k \parallel f \parallel^2, \quad \forall f \epsilon D(A^*) \end{aligned} \tag{2.22}$$

Proof:

Condition [2.22] becomes [19]:

$$< \lambda x - Ax, \lambda x - Ax > \geq (\lambda - \omega)^2 < x, x > \text{ for } \lambda > \omega, x \epsilon D(A) \tag{2.23}$$

which gives:

$$2\lambda \left(\omega \|x\|^2 - Re < Ax, x > + < Ax, x > - \omega^2 \|x\|^2\right) \geq 0 \tag{2.24}$$

this will hold if we can find a constant β such that:

$$\begin{aligned} \beta \|x\|^2 &\geq Re(< Ax, x >), \ x \epsilon D(A) \\ \beta \|x\|^2 &\geq Re(< A^* x, x >), \ x \epsilon D(A^*) \end{aligned} \tag{2.25}$$

With these theorems from semi-group-theory available to us, we will show that $A = \begin{bmatrix} 0 & 1 \\ -L & c \end{bmatrix}$ generates a C_0 semi-group.

We have shown that the differential operator L is self-adjoint and therefore we can define a square root $L^{1/2}$ by means of functional calculus of unbounded operators. In general $L^{1/2}$ cannot be identified in terms of L and may even fail to be a differential operator, however it domain turms out to be:

$$D(L^{1/2}) = \overset{\circ}{H}_1 (\Omega) \tag{2.26}$$

(For a proof see page 105 in Fattorini [25])

Let us introduce the Hilbert space:

$$H = D(L^{\frac{1}{2}}) \times L^2(\Omega) \tag{2.27}$$

For $f = \begin{bmatrix} u \\ v \end{bmatrix}, f' = \begin{bmatrix} u' \\ v' \end{bmatrix} \epsilon H$ we define the inner product as follows:

$$< f, f' >_H = (L^{\frac{1}{2}} u, L^{\frac{1}{2}} u')_{L^2(\Omega)} + (v, v')_{L^2(\Omega)} \tag{2.28}$$

Now let $f' = Af = \begin{bmatrix} v \\ -Lu + cv \end{bmatrix}$

$$\begin{aligned} < f, Af >_H &= (L^{\frac{1}{2}} u, L^{\frac{1}{2}} v)_{L^2(\Omega)} + (v, -Lu + cv)_{L^2(\Omega)} \\ &= (Lu, v)_{L^2(\Omega)} - (v, Lu)_{L^2(\Omega)} + c \parallel v \parallel^2_{L^2(\Omega)} \\ &= c \parallel v \parallel^2_{L^2(\Omega)} \end{aligned} \tag{2.29}$$

for $f \epsilon D(L^{1/2}) \times D(L) \subset H$.

By similar computations: $A^* = \begin{bmatrix} 0 & -1 \\ L & c \end{bmatrix} \Rightarrow D(A^*) = D(A)$ and $\langle f, A^* f \rangle_H = c \parallel v \parallel^2_{L^2(\Omega)}$. Applying this theorem and the above analysis we conclude that:

Theorem 2.4:

The operator A is the infinitesimal generator of a strongly continuous semigroup T_t. Moreover if the condition:

$$\pi^2 \left(\frac{a^2}{\ell_x^2} + \frac{b^2}{\ell_y^2} \right) - \left(d + \frac{c^2}{4} \right) > 0 \tag{2.30}$$

is satisfied T_t will be exponentially stable.

Proof:

(a) by the previous analysis and corollary [2.3] A is the generator of a C_0 semigroup T_t.

(b) Since $A = \begin{bmatrix} 0 & 1 \\ -L & c \end{bmatrix}$ it is a simple matter to check that the eigenvalues Λ of A are related to the eigenvalues λ of L by the second order equation:

$$\Lambda^2 - c\Lambda + \lambda = 0 \tag{2.31}$$

which admits the solution:

$$\Lambda = \frac{c \pm \sqrt{c^2 - 4\lambda}}{2} \tag{2.32}$$

If $c^2 - 4\lambda < 0$ then $Re(\Lambda) = \frac{c}{2} < 0$.

Since the eigenvalues λ of the positive second order idfferential operator L tend to infinity, it is enough to satisfy the condition

$$c^2 - 4\lambda_{\text{min}} < 0 \tag{2.33}$$

where λ_{min} is the smallest eigenvalue of L, and this condition is equivalent to the one given in the hypothesis of the theorem. Now Let $\sigma(A)$ be the spectrum of A. The condition $Re(\Lambda) = \frac{c}{2}$ implies that:

$$Re(\sigma(A)) = \frac{c}{2} < 0 \tag{2.34}$$

and by the spectrum determined growth assumption (see references [19, 68]) we can choose ω_0 in Theorem [2.2] to satisfy:

$$\frac{c}{2} < \omega_0 < 0 \tag{2.35}$$

and hence the exponential stability of the semi-group.

2.1.2 Numerical Integration by the Method of Lines

There are various approaches to the numerical integration of partial differential equations. Some of the most important methods in this respect are:

(a) Finite difference methods;

(b) Finite element methods;

(c) Characteristics (for hyperbolic equations);

(d) "Exact" methods (mainly series expansion of the solution).

It would take us too long to go into the main differences between all these methods, the range of their applicability and their respective performance.

In this subsection we intend to apply the method of lines to get a numerical solution of the PDE (2.3). This method consists of defining a grid on the domain of interest and then transforming the PDE into a system of ODE's. Subsequently, one uses a good ODE integrator (one that handles discontinuities, stiffness, etc, successfully).

Consider the equation of the vibrating membrane:

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} = a\frac{\partial^2 h}{\partial x^2} + b\frac{\partial^2 h}{\partial y^2} + c\frac{\partial h}{\partial t} + dh + f \quad \text{in} \quad \overset{\circ}{\Omega} \\ h|_{\partial\Omega} = 0 \\ h = h_t = 0 \quad \text{at} \quad t = 0 \end{cases} \tag{2.36}$$

Let us make the following change of variables:

$$\begin{cases} p = h \\ q = h_t \end{cases} \tag{2.37}$$

Then (2.22) is equivalent to the following 1^{st} order (in t) system:

$$\begin{cases} \frac{\partial p}{\partial t} = q \\ \frac{\partial q}{\partial t} = a\frac{\partial^2 p}{\partial x^2} + b\frac{\partial^2 p}{\partial y^2} + cq + dp + f \\ p = q = 0 \quad \text{on} \quad \partial\Omega \\ p = q = 0 \quad \text{at} \quad t = 0 \end{cases} \tag{2.38}$$

Next we devise a semidiscretization algorithm for (2.24). To do this we introduce a grid of points (x_i, y_i) on the region Ω:

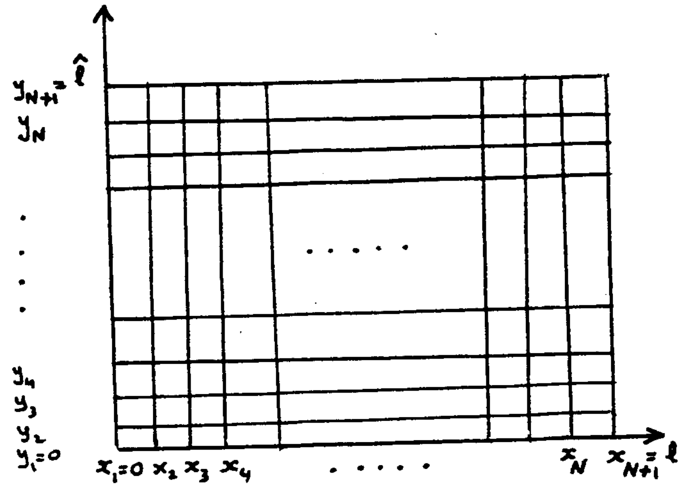


Fig. 1: Subdivision of the Membrane

We discretize the space variables using the central difference schemes for the second derivative (which is of order h^2):

$$\frac{\partial^2 p}{\partial x^2}\,|_{i,j} = \frac{p_{i-1,j} - 2p_{i,j} + p_{i+1,j}}{h^2}$$

$$\frac{\partial^2 p}{\partial y^2}\,|_{i,j} = \frac{p_{i,j-1} - 2p_{i,j} + p_{i,j+1}}{k^2} \tag{2.39}$$

with the obvious notation:

$$p_{ij} = p(x_i, y_i, t) \tag{2.40}$$

$$h = \frac{\ell}{N}, k = \frac{\hat{\ell}}{N} \tag{2.41}$$

Thus we get:

For $i = 2, N; j = 2, N$:

$$\begin{cases} \frac{dp_{ij}}{dt} = q_{ij} \\ \frac{dq_{ij}}{dt} = a \cdot \frac{p_{i-1,j} - 2p_{ij} + p_{i+1,j}}{h^2} + b\frac{p_{i,j-1} - 2p_{ij} + p_{i,j+1}}{k^2} + cq_{ij} + dp_{ij} + f_{ij} \\ p_{ij}(0) = q_{ij}(0) = 0 \end{cases} \tag{2.42}$$

For $i = 1, \cdots N+1 \begin{cases} j = 1 \\ \text{or } j = N+1 \end{cases}$

and $j = 1, \cdots N+1 \begin{cases} j = 1 \\ \text{or } i = N+1 \end{cases}$ We get:

$$p_{ij} = q_{ij} = 0 \tag{2.43}$$

The system (2.38) can be rewritten in the following form:

$$\begin{cases} p'_{ij}(t) = q_{ij} \\ q'_{ij}(t) = \frac{a}{h^2}(p_{i-1,j} + p_{i+1,j}) + \frac{b}{k^2}(p_{i,j-1} + p_{i,j+1}) \\ \qquad + [d - 2(\frac{a}{h^2} + \frac{b}{k^2})]p_{ij} + cq_{ij} + f_{ij} \\ p_{ij}(0) = q_{ij}(0) = 0 \end{cases} \tag{2.44}$$

Let us convert P_{ij}, q_{ij} into a single one dimensional array.

Case 1

$$\begin{array}{ll} \underline{i = 1, \cdots N+1} & \text{Let } p_{ij} = y((i-1)(N+1)+j) \\ \underline{j = 1, \cdots N+1} & \end{array} \tag{2.45}$$

Case 2

$$\begin{array}{ll} \underline{i = 1, \cdots N+1} & \text{Let } q_{ij} = y\{(N+1)^2 + (i-1)(N+1) + j\} \\ \underline{j = 1, \cdots N+1} & \qquad q_{ij} = y\{(N+1)(N+i) + j\} \end{array} \tag{2.46}$$

So we end up with the one dimensional array y, which relates to p_{ij}, q_{ij} as follows:

$$\begin{array}{ll} p_{ij} = y\{(i-1)(N+1)+j\} & i,j = 1, \cdots N+1 \\ q_{ij} = y\{(N+1)(N+i)+j\} & i,j = 1, \cdots N+1 \end{array} \tag{2.47}$$

Thus the method of lines leads to the integration of $\underline{2(N-1)^2}$ ordinary differential equations, where N is the number of grid-points. The method is therefore time-consuming for large N. An additional drawback of the method (as implemented here) seems to be a stability problem: the solution diverges after a relatively small number of steps (~ 5) (for a study of convergence of this method see [65].

Therefore we were led to look for a more satisfying method of solving the hyperbolic system in question and the eigenfunction expansion method gives superior numerical results and converges very fast. This latter approach will be the subject of the next paragraph.

2.1.3 Eigenfunction Expansion of the Solution to the Hyperbolic System

Consider the equation of the vibrating membrane:

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} = a\frac{\partial^2 h}{\partial \alpha^2} + b\frac{\partial^2 h}{\partial \beta^2} + c\frac{\partial h}{\partial t} + dh + u \text{ in } \Omega \\ h|_{\partial\Omega} = 0 \\ h(0), h'(0) \quad \text{given} \end{cases} \tag{2.48}$$

Let

$$g(t) = h(t)e^{-\frac{c}{2}t} \tag{2.49}$$

This change of dependent variable allows us to get rid-of the damping term $a\frac{\partial h}{\partial t}$; however the d-coefficient will change, and the new control will become:

$$v(t) = u(t)e^{-\frac{c}{2}t} \tag{2.50}$$

We now get the new system:

$$\begin{cases} \frac{\partial^2 g}{\partial t^2} = a\frac{\partial^2 g}{\partial x^2} + b\frac{\partial^2 g}{\partial y^2} + (d + \frac{c^2}{4})g + v \\ g|_{\partial\Omega} = 0 \\ g(0) = h(0) \\ g'(0) = h'(0) - \frac{c}{2}h(0) \end{cases} \tag{2.51}$$

Here we see that the coefficient of g has changed ($d \rightarrow d + \frac{c^2}{4}$) and also the initial condition for the velocity has changed ($h'(0) \rightarrow h'(0) - \frac{c}{2}h(0)$) the control being $v = e^{-\frac{c}{2}t}u$. These remarks are crucial for the simulation of the original system. However it is equivalent to study the following equation, while making the necessary adjustments for the coefficients:

$$\begin{cases} \frac{\partial^2 g}{\partial t^2} = a\frac{\partial^2 g}{\partial x^2} + b\frac{\partial^2 g}{\partial y^2} + dg + bu \quad \text{in} \quad \mathring{\Omega} \\ g = 0 \quad \text{on} \quad \partial\Omega \\ g(x,y,t_0) = g_0(x,y) \\ g_t(x,y,t_0) = v_0(x,y) \end{cases} \tag{2.52}$$

Here we do not assume $t_0 = 0$. We also assume that we have the general initial conditions g_0 and v_0, because we will use the solution of this system to simulate the response of the membrane.

To solve the above we use the classical technique of eigenfunction expansion to get:

(a) an eigenvalue problem of a partial differential operator.

(b) a system of ordinary differential equations.

Let $g(x,y,t) = \sum_{m,n} \alpha_{mn}(t)\varphi_{mn}(x,y)$ where $\varphi_{mn}(x,y)$ are the eigenfunctions of the associated partial differential operator.

$$u(x,y,t) = \sum_{m,n} u_{mn}(t)\varphi_{mn}(x,y) \tag{2.53}$$

where $u(x,y,t)$ is the control acting on the membrane. We decompose the initial conditions in the same way:

$$g(x,y,t_0) = \sum_{m,n} r_{mn}\varphi_{mn}(x,y)$$

$$g_t(x,y,t_0) = \sum_{m,n} v_{mn}\varphi_{mn}(x,y) \tag{2.54}$$

Plugging the formulas (2.53) and (2.54) into the system (2.52) and differentiating the series term by term ("formally") as required by the equations we get:

(a) Eigenvalue problem:

$$\begin{cases} a\frac{\partial^2 \varphi_{mn}}{\partial x^2} + b\frac{\partial^2 \varphi_{mn}}{\partial y^2} + d\varphi_{mn} = \lambda_{mn}\varphi_{mn} \\ \varphi_{mn}|_{\partial\Omega} = 0 \end{cases} \tag{2.55}$$

(b) Ordinary initial value problem:

$$\begin{cases} \alpha''_{mn}(t) = \lambda_{mn}\alpha_{mn}(t) + bu_{mn}(t) \\ \alpha_{mn}(0) = r_{mn} \\ \alpha'_{mn}(0) = v_{mn} \end{cases} \tag{2.56}$$

To solve the eigenvalue problem we use standard techniques to get:

$$\lambda_{mn} = d - \left\{ a\left(\frac{n\pi}{\ell}\right)^2 + b\left(\frac{m\pi}{\hat{\ell}}\right)^2 \right\}$$

$$\varphi_{mn}(x,y) = \sin\left(\frac{n\pi x}{\ell}\right) \cdot \sin\left(\frac{m\pi y}{\hat{\ell}}\right) \tag{2.57}$$

The original initial value problem can be transformed into the following first order system of ODE's:

$$\begin{cases} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & \lambda \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ b \end{bmatrix} u \\ x_1(t_0) \text{ given} \\ x_2(t_0) \text{ given} \end{cases} \tag{2.58}$$

If we define $A = \begin{bmatrix} 0 & 1 \\ \lambda & 0 \end{bmatrix}$ then, $\mid sI - A \mid = s^2 - \lambda$ is the characteristic polynomial.

There are three cases:

(1) $\lambda < 0$

(2) $\lambda = 0$

(3) $\lambda > 0$

(3) can be deduced from (1) by using the complex notation (2) has a straightforward solution, and only the solution of (1) needs to be computed.

(1) $\underline{\lambda < 0}$

The solution of (2.58) in terms of $\mu = \sqrt{|\lambda|}$ is given by: which leads to the solution:

$$\alpha(t) = r\cos\mu t + \frac{v}{\mu}\sin\mu t + \frac{b}{\mu}\int_0^t u(\tau)\sin\mu(t-\tau)d\tau$$

$$\alpha'(t) = -\mu r\sin\mu t + v\cos\mu t + b\int_0^t u(\tau)\cos\mu(t-\tau)d\tau \tag{2.59}$$

We have dropped the dependence of μ, r, v, u on the mode (m, n) for notational convenience only.

It is also very important to keep the expression of $\alpha'(t)$ (which can be obtained by direct differentiation of the formula for $\alpha(t)$) to be able to compute the velocity of the membrane directly. Hence all the problems associated with numerical differentiation are avoided.

(2) $\lambda = 0$

A straightforward computation gives:

$$\alpha_{mn}(t) = r_{mn} + v_{mn}t + \int_0^t u_{mn}(\tau)(t-\tau)d\tau$$

$$\alpha'_{mn}(t) = v_{mn} + \int_0^t u_{mn}(\tau)\tau d\tau \tag{2.60}$$

(3) $\lambda > 0$

Defining $\mu = \sqrt{\lambda}$, we get the solution of (2.48) corresponding to this case:

$$\alpha_{mn}(t) = r_{mn} ch\mu_{mn}t + \frac{v_{mn}}{\mu_{mn}} sh\mu_{mn}t + \frac{b}{\mu_{mn}} \int_0^t u_{mn}(\tau) sh\mu_{mn}(t-\tau)d\tau$$

$$\alpha'_{mn}(t) = r_{mn}\mu_{mn} sh\mu_{mn}t + v_{mn} ch\mu_{mn}t + b\int_0^t u_{mn}(\tau) ch\mu_{mn}(t-\tau)d\tau \tag{2.61}$$

This case is completely symmetrical to case (1) and can be obtained by changing μ to $i\mu$.

2.1.4 Numerical Simulation

Using eigenfunction expansions we have reduced the solution to the PDE (2.3) to the computation of one dimensional integrals as given by equations (2.59, 2.60, 2.61).

Here we give five examples:

Example I: a vibrating free membrane

Example II: a vibrating free membrane with damping.

Example III: a membrane with damping and subject to a forcing term.

Example IV: this example does not represent a "physical" membrane.

Example V: this example involves the full "generalized wave" equation
with a forcing term containing exponential, harmonic
and constant terms.

In all these examples the exact solution is compared to the computed solution as given by equations (2.53, 2.54, 2.59, 2.60, 2.61), but for convenience only example V results are reported.

Example I (Oscillating free membrane)

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} &= v^2 \left(\frac{\partial^2 h}{\partial x^2} + \frac{\partial^2 h}{\partial y^2} \right) \quad \text{in } \Omega \\ h|_{\partial\Omega} &= 0 \\ h(0) &= h_0 \\ h'(0) &= 0 \end{cases} \tag{2.62}$$

Let

$$h_0\,(x,y) = k\; sin\, \left(\frac{n_0\,\pi}{\ell} x\right)\; sin\, \left(\frac{m_0\,\pi}{\hat{\ell}} y\right)$$

$$\lambda = \left[\left(\frac{n_o\,\pi}{\ell}\right)^2 + \left(\frac{m_0\,\pi}{\hat{\ell}}\right)^2 \right]^{1/2}$$

$$\varphi(x,y,t) = h_0(x,y)\; cos(\lambda_{m_0 n_0}\; vt)$$

$$\psi(t) = h_0(x,y)\; sin(\lambda_{m_0 n_0}\; vt) \tag{2.63}$$

Then the solution of problem (2.62) is:

$$\begin{cases} h(x,y,t) = \varphi(x,y,t) \\ h'(x,y,t) = -\lambda v \psi(x,y,t) \end{cases} \tag{2.64}$$

Example II:

(Free system with damping)

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} = v^2 \ \left(\frac{\partial^2 h}{\partial x^2} + \frac{\partial^2 h}{\partial y^2}\right) - \frac{c^2}{4} h + c\frac{\partial h}{\partial t} \\ h(0) = h_0 = h \sin\left(\frac{n_0 \lambda}{\ell} x\right) \sin\left(\frac{n_0 \lambda}{\ell} y\right) \\ h'(0) = \frac{c}{2} h_0 \\ h|_{\partial\Omega} = 0 \end{cases} \tag{2.65}$$

The solution is given by:

$$\begin{cases} h(t) \ = \ e^{\frac{c}{2}t} cos(\lambda_{m_0 n_0} vt) \cdot k \cdot sin\left(\frac{n_0 \pi}{\ell} x\right) \cdot sin\left(\frac{m_0 \pi}{\ell} y\right) \\ h'(t) \ = \ e^{\frac{c}{2}t} \left(\frac{c}{2} cos(\lambda_{m_0 n_0} vt) \ - \ \lambda_{m_0 n_0} v \cdot sin(\lambda_{m_0 n_0} vt)\right) \\ \qquad \cdot k \ sin\left(\frac{n_0 \pi}{\ell} x\right) sin\left(\frac{m_0 \pi}{\ell} y\right) \end{cases} \tag{2.66}$$

Example III

(Membrane with damping subject to a forcing term)

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} \ = \ v^2 \left(\frac{\partial^2 h}{\partial x^2} \ + \ \frac{\partial^2 h}{\partial y^2}\right) \ - \ \frac{c^2}{4} \ h \ + \ c \ \frac{\partial h}{\partial t} \ + \ u \\ h(0) \ = \ h'(0) \ = \ 0 \\ h|_{\partial\Omega} \ = \ 0 \end{cases} \tag{2.67}$$

The solution of system (2.67) is:

$$\begin{cases} h(x,y,t) \ = \ e^{\frac{c}{2}t} \ (\cos(\lambda vt) - 1) \cdot k \cdot sin\left(\frac{n_0 \pi}{\ell} x\right) \cdot sin\left(\frac{m_0 \pi}{\ell} y\right) \\ h'(x,y,t) \ = \ \frac{c}{2} \ h(x,y,t) \ - \ e^{\frac{c}{2}t} \cdot \lambda v \cdot sin(\lambda vt) k \ sin\left(\frac{n_0 \pi}{\ell} x\right) sin\left(\frac{m_0 \pi}{\ell} y\right) \end{cases} \tag{2.68}$$

Example IV

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} \ = \ d \cdot h \\ h(0) \ = \ k \ sin\left(\frac{n_0 \pi}{\ell} x\right) \cdot sin\left(\frac{m_0 \pi}{\ell} y\right) \\ h'(0) \ = \ 0 \\ h|_{\partial\Omega} \ = \ 0 \end{cases} \tag{2.69}$$

Define $\mu = \sqrt{|d|}$

Case 1: $d > 0$

$$\begin{cases} h(x,y,t) \;=\; k\; ch\mu t\; sin\left(\frac{n_0\pi}{\ell}x\right)\; sin\;\left(\frac{m_0\pi}{\hat{\ell}}y\right) \\ h'(x,y,t) \;=\; k\;\mu\; sh\mu t\; sin\left(\frac{n_0\pi}{\ell}x\right)\; sin\;\left(\frac{m_0\pi}{\hat{\ell}}y\right) \end{cases} \tag{2.70}$$

Case 2: $d < 0$

$$\begin{cases} h(x,y,t) \;=\; k\; sin\mu t\;\cdot sin\left(\frac{n_0\pi}{\ell}x\right)\; sin\;\left(\frac{m_0\pi}{\hat{\ell}}y\right) \\ h'(x,y,t) \;=\; k\;\mu\; cos\mu t\;\cdot sin\left(\frac{n_0\pi}{\ell}x\right)\; sin\;\left(\frac{m_0\pi}{\hat{\ell}}y\right) \end{cases} \tag{2.71}$$

Example V

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} \;=\; a\,\frac{\partial^2 h}{\partial x^2} \;+\; b\,\frac{\partial^2 h}{\partial y^2} \;+\; w\,\frac{\partial h}{\partial t} \;+\; dh \;+\; u \\ h|_{\partial\Omega} \;=\; 0 \\ h(0) \;=\; (A \;+\; B)\,\varphi_{mn} \;+\; (C \;+\; D)\,\varphi_{k\ell} \\ h'(0) \;=\; A\,\varphi_{mn} \end{cases} \tag{2.72}$$

Let

$$\lambda_{mn} \;=\; -\;\left\{a(\frac{n\pi}{L})^2 \;+\; b\,(\frac{m\pi}{\hat{L}})^2\right\}$$

$$\lambda_{kl} \;=\; -\;\left\{a(\frac{k\pi}{L})^2 \;+\; b\,(\frac{\ell\pi}{\hat{L}})^2\right\} \tag{2.73}$$

Let the control be given by:

$$\begin{aligned} u(x,y,t) \;=\; & \left\{Ae^t[1-(\lambda_{mn} \;+\; w \;+\; d)] - B(\lambda_{mn} \;+\; d)\right\}\varphi_{mn} \\ & + \left\{-C\cos t(1+\lambda_{k\ell}+d) + cw\sin t - D(\lambda_{k\ell}+d)\right\}\varphi_{k\ell} \end{aligned} \tag{2.74}$$

Then the solution to the wave equation is given by:

$$\begin{cases} h(t) \;=\; (Ae^t \;+\; B)\,\varphi_{mn} \;+\; (C\; cos\; t \;+\; D)\varphi_{h\ell} \\ h'(t) \;=\; Ae^t\,\varphi_{mn} \;-\; C\; sin\; t\;\varphi_{h\ell} \end{cases} \tag{2.75}$$

φ_{mn} and $\varphi_{k\ell}$ are the eigenfunctions corresponding to eigenvalues λ_{mn} & $\lambda_{k\ell}$.

Next we report the results of our simulation and we would like to explain to the reader what was done and how to read the tables.

- The equations (2.59, 2.60, 2.61) have been implemented using an adaptive integrator and we need to provide it with the absolute and relative errors on the solution: $0.1, 0.0$.
- The control as given by equation (2.74) is sampled every T seconds and we need to provide the program with T and the total number of periods N: $T = 0.1, N = 20$.
- We also need to provide the parameters of the membrane a, b, c, d: $a = 1.2, b = 2.3, c = -1.7, d = .6$.
- The coefficients A, B, C, D are: $A = .75, B = 1.2, C = -2.4, D = .01$.
- The number of grid points (see Fig. 1) and the dimensions of the membrane are: $N = 20, \ell = 1.5, \hat{\ell} = 2.3$.
- Finally the index of the corresponding eigenfunction is: $m = n = 1$.

The output of the program (is the "image" of the membrane) is organized as follows:

- the time is printed
- the left hand matrix gives the displacement as computed from equation (2.54) at the desired mesh points (see Fig. 1) on the membrane.
- the right hand matrix gives the exact displacement as given by equations (2.75) at the desired mesh point (see Fig. 1) on the membrane.
- similary for the velocity.

As we can see from the numerical results the eigenfunction expansion gives an excellent method for computing the solution to the "Generalized" wave equation, it also permits the computation of the velocity without numerical differentiation. Another feature of the simulation algorithm is that it "preserves" the symmetry of the problem: neutral lines and symmetries with respect to these lines are preserved, and this is very important for the control alogrithm.

As we mentioned time and again, very few terms are necessary in the eigenfunction expansion (few mode shapes for physical systems) even though the LSS usually comprises thousands of elements, which makes our approach very attractive.

Time = 0

Displacement					Exact Displacement				
-1.4671	-1.5300	-1.5511	-1.5300	-1.4671	-1.4671	-1.5300	-1.5511	-1.5300	-1.4671
-1.5300	-1.5956	-1.6176	-1.5956	-1.5300	-1.5300	-1.5956	-1.6176	-1.5956	-1.5300
-1.5511	-1.6176	-1.6400	-1.6176	-1.5511	-1.5511	-1.6176	-1.6400	-1.6176	-1.5511
-1.5300	-1.5956	-1.6176	-1.5956	-1.5300	-1.5300	-1.5956	-1.6176	-1.5956	-1.5300
-1.4671	-1.5300	-1.5511	-1.5300	-1.4671	-1.4671	-1.5300	-1.5511	-1.5300	-1.4671
Velocity					**Exact Velocity**				
1.0824	1.1288	1.1444	1.1288	1.0824	1.0824	1.1288	1.1444	1.1288	1.0824
1.1288	1.1772	1.1935	1.1772	1.1288	1.1288	1.1772	1.1935	1.1772	1.1288
1.1444	1.1935	1.2100	1.1935	1.1444	1.1444	1.1935	1.2100	1.1935	1.1444
1.1288	1.1772	1.1935	1.1772	1.1288	1.1288	1.1772	1.1935	1.1772	1.1288
1.0824	1.1288	1.1444	1.1288	1.0824	1.0824	1.1288	1.1444	1.1288	1.0824

Time = 0.02

Displacement					Exact Displacement				
-1.4450	-1.5069	-1.5278	-1.5069	-1.4450	-1.4450	-1.5070	-1.5278	-1.5070	-1.4450
-1.5069	-1.5715	-1.5933	-1.5715	-1.5069	-1.5070	-1.5716	-1.5933	-1.5716	-1.5070
-1.5278	-1.5933	-1.6153	-1.5933	-1.5278	-1.5278	-1.5933	-1.6153	-1.5933	-1.5278
-1.5069	-1.5715	-1.5933	-1.5715	-1.5069	-1.5070	-1.5716	-1.5933	-1.5716	-1.5070
-1.4450	-1.5069	-1.5278	-1.5069	-1.4450	-1.4450	-1.5070	-1.5278	-1.5070	-1.4450
Velocity					**Exact Velocity**				
1.1264	1.1747	1.1910	1.1747	1.1264	1.1255	1.01738	1.1900	1.1738	1.1255
1.1747	1.2251	1.2420	1.2251	1.1747	1.738	1.2241	1.2410	1.2241	1.1738
1.1910	1.2420	1.2592	1.2420	1.1910	1.1900	1.2410	1.2582	1.2410	1.1900
1.1747	1.2251	1.2420	1.2251	1.1747	1.738	1.2241	1.2410	1.2241	1.1738
1.1264	1.1747	1.1910	1.1747	1.1264	1.1255	1.01738	1.1900	1.1738	1.1255

Fig. 2: Exact Displacement/Velocity of the Membrane Compared With the Computed Values by Our Algorithm (Example V)

Time = 0.04

Displacement					Exact Displacement				
-1.4220	-1.4830	-1.5035	-1.4830	-1.4220	-1.4221	-1.4830	-1.5035	-1.4830	-1.4221
-1.4830	-1.5466	-1.567	-1.5466	-1.4830	-1.4830	-1.5466	-1.5680	-1.5466	-1.4830
-1.5035	-1.5679	-1.5896	-1.5679	-1.5035	-1.5035	-1.5680	-1.5897	-1.5680	-1.5035
-1.4830	-1.5466	-1.567	-1.5466	-1.4830	-1.4830	-1.5466	-1.5680	-1.5466	-1.4830
-1.4220	-1.4830	-1.5035	-1.4830	-1.4220	-1.4221	-1.4830	-1.5035	-1.4830	-1.4221
Velocity					**Exact Velocity**				
1.1708	1.2210	1.2379	1.2210	1.1708	1.1687	1.2187	1.2356	1.2187	1.1687
1.2210	1.2733	1.2909	1.2733	1.2210	1.2187	1.2710	1.2886	1.2710	1.2187
1.2379	1.2909	1.3088	1.2909	1.2379	1.2356	1.2886	1.3064	1.2886	1.2356
1.2210	1.2733	1.2909	1.2733	1.2210	1.2187	1.2710	1.2886	1.2710	1.2187
1.1708	1.2210	1.2379	1.2210	1.1708	1.1687	1.2187	1.2356	1.2187	1.1687

Time = 0.05

Displacement					Exact Displacement				
-1.4102	-1.4707	-1.4910	-1.4707	-1.4102	-1.4103	-1.4707	-1.4911	-1.4707	-1.4103
-1.4707	-1.5337	-1.5549	-1.5337	-1.4707	-1.04707	-1.5338	-1.5550	-1.5338	-1.4707
-1.4910	-1.5549	-1.5764	-1.5549	-1.4940	-1.4911	-1.5550	-1.5765	-1.5550	-1.4911
-1.4707	-1.5337	-1.5549	-1.5337	-1.4707	-1.04707	-1.5338	-1.5550	-1.5338	-1.4707
-1.4102	-1.4707	-1.4910	-1.4707	-1.4102	-1.4103	-1.4707	-1.4911	-1.4707	-1.4103
Velocity					**Exact Velocity**				
1.1930	1.2441	1.2613	1.2441	1.1930	1.1902	1.2412	1.2584	1.2412	1.1902
1.2441	1.2975	1.3154	1.2975	1.2441	1.2412	1.2944	1.3123	1.2944	1.2412
1.2631	1.3154	1.3336	1.3154	1.2613	1.2584	1.3123	1.3305	1.3123	1.2584
1.2441	1.2975	1.3154	1.2975	1.2441	1.2412	1.2944	1.3123	1.2944	1.2412
1.1930	1.2441	1.2613	1.2441	1.1930	1.1902	1.2412	1.2584	1.2412	1.1902

Fig. 2 (Cont'd): Exact Displacement/Velocity of the Membrane Compared With the Computed Values by Our Algorithm (Example V)

Time = 0.07

Displacement					Exact Displacement				
-1.3858	-1.4452	-1.4652	-1.4452	-1.3858	-1.3860	-1.4455	-1.4654	-1.4455	-1.3860
-1.4452	-1.5071	-1.5280	-1.5071	-1.4452	-1.4455	-1.5074	-1.5283	-1.5074	-1.4455
-1.4652	-1.5280	-1.5491	-1.5280	-1.4652	-1.4654	-1.5283	-1.5494	-1.5283	-1.4654
-1.4452	-1.5071	-1.5280	-1.5071	-1.4452	-1.4455	-1.5074	-1.5283	-1.5074	-1.4455
-1.3858	-1.4452	-1.4652	-1.4452	-1.3858	-1.3860	-1.4455	-1.4654	-1.4455	-1.3860
Velocity					Exact Velocity				
1.2373	1.2904	1.3082	1.2904	1.2373	1.2332	1.2861	1.3039	1.2861	1.2332
1.2904	1.3457	1.3643	1.3457	1.2904	1.2861	1.3412	1.3598	1.3412	1.2861
1.3082	1.3643	1.3831	1.3643	1.3082	1.3039	1.3598	1.3786	1.3598	1.3039
1.2904	1.3457	1.3643	1.3457	1.2904	1.2861	1.3412	1.3598	1.3412	1.2861
1.2373	1.2904	1.3082	1.2904	1.2373	1.2332	1.2861	1.3039	1.2861	1.2332

Time = 0.09

Displacement					Exact Displacement				
-1.3607	-1.4190	-1.4387	-1.4190	-1.3607	-1.3609	-1.4193	-1.4389	-1.4193	-1.3609
-1.4190	-1.4799	-1.5003	-1.4799	-1.4190	-1.4193	-1.4801	-1.5006	-1.4801	-1.4193
-1.4387	-1.5003	-1.5211	-1.5003	-1.4387	-1.4389	-1.5006	-1.5213	-1.5006	-1.4389
-1.4190	-1.4799	-1.5003	-1.4799	-1.4190	-1.4193	-1.4801	-1.5006	-1.4801	-1.4193
-1.3607	-1.4190	-1.4387	-1.4190	-1.3607	-1.3609	-1.4193	-1.4389	-1.4193	-1.3609
Velocity					Exact Velocity				
1.2817	1.3366	1.3551	1.3366	1.2817	1.2762	1.3309	1.3494	1.3309	1.2762
1.3366	1.3939	1.4132	1.3939	1.3366	1.3309	1.3880	1.4072	1.3880	1.3309
1.3551	1.4132	1.4327	1.4132	1.3551	1.3494	1.4072	1.4267	1.4072	1.3494
1.3366	1.3939	1.4132	1.3939	1.3366	1.3309	1.3880	1.4072	1.3880	1.3309
1.2817	1.3366	1.3551	1.3366	1.2817	1.2762	1.3309	1.3494	1.3309	1.2762

Fig. 2 (Cont'd): Exact Displacement/Velocity of the Membrane Compared With the Computed Values by Our Algorithm (Example V)

2.2 Study Of The Elliptic System

In this chapter we apply the Fourier transform, with respect to the time variable, to the Wave equation. This leads us to an elliptic system we study using the Green operator technique.

We then expand the real Green's function into an exponentially converging series. A first order discretization of the analytic expression for the Green's function allows us to find an approximate solution to the elliptic system.

Next we consider the system with damping and use the analytic continuation principle to compute the "complex" Green's function. Several special cases are given and the numerical implementation is discussed.

2.2.1 Fourier Transform of the Wave Equation

Consider the Wave equation:

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} = a \, \frac{\partial^2 h}{\partial x^2} + b \, \frac{\partial^2 h}{\partial y^2} + c \, \frac{\partial h}{\partial t} + dh + u \\ h|_{\partial\Omega} = 0 \end{cases} \tag{2.76}$$

Applying the Fourier transform with respect to t gives:

$$\begin{cases} a \, \frac{\partial^2 H}{\partial x^2} + b \, \frac{\partial^2 H}{\partial y^2} + (d + \omega^2 - jc\omega) \, H = -u \\ H|_{\partial\Omega} = 0 \end{cases} \tag{2.77}$$

If we define $z(\omega) = d + \omega^2 - jc\omega \in \mathbb{C}$, the elliptic equation becomes:

$$\begin{cases} a \, \frac{\partial^2 H}{\partial x^2} + b \, \frac{\partial^2 H}{\partial y^2} + z(\omega) \, H = u \\ H|_{\partial\Omega} = 0 \end{cases} \tag{2.78}$$

Define the operator $L_z = a\frac{\partial^2}{\partial x^2} + b\frac{\partial^2}{\partial y^2} + z(\omega)$ to get the following Dirichlet problem;

$$\begin{cases} L_z \, u = f \qquad \text{in } \mathring{\Omega} \\ u|_{\partial\Omega} = 0 \end{cases} \tag{2.79}$$

Using the same argument as in the previous section, we conclude that L_z is self-adjoint i.e. $L_z^* = L_z$. However, L_z has complex coefficients and the eigenvalues of L_z are those of L shifted by z. In other words, the presence of the damping c shifts the poles of the Green's function from the real axis making it possible to use usual integration techniques.

Note: Although $L_z^* = L_z$ we may use L_z^* instead of L_z to conform to the PDE literature.

Consider $C_c^\infty(\Omega)$ the space of test functions with compact support on Ω and define the bilinear form:

$$B[u,v] = (u,\ L_z^* v) \qquad \forall v \in C_c^\infty\ (\Omega) \tag{2.80}$$

This leads to the weak form of the Dirichlet problem:

For $f \in L_2(\Omega)$, find $u \in L_2^{\mathrm{loc}}(\Omega)$ such that:

$$B[u,v] = (f,v) \qquad \forall v \in C_c^\infty(\Omega) \tag{2.81}$$

u is a weak solution of the Dirichlet problem.

If we assume $u \in H_1(\Omega_o), \bar{\Omega}_0 \subset \Omega$ and

$$B[u,v] = (f,v) \tag{2.82}$$

then u is a strong solution.

If $u \in C^2(\Omega)$ then u is a classical solution. In fact, in this special case it is obvious that $u|_{\partial\Omega} = 0$ is the trace of the function $u \equiv 0 \ \in \overset{\circ}{H}_1\ (\Omega)$.

The Generalized Dirichlet problem becomes:

Find $u \in \overset{\circ}{H}_1\ (\Omega)$ such that:

$$B[u,v] = (f,v),\ \forall v \in \overset{\circ}{H}_1\ (\Omega) \tag{2.83}$$

Notice that the right hand side (f, v) is a continuous linear function F on $\overset{\circ}{H}_1 (\Omega)$, which leads to the new formulation of the Dirichlet problem:

Find $u \epsilon \overset{\circ}{H}_1 (\Omega)$ such that:

$$B[u,\ v] \ = \ F(v), \forall v \epsilon \ \overset{\circ}{H}_1 (\Omega) \tag{2.84}$$

where $F \epsilon H_{-1}(\Omega)$.

Case 1: $z \ = \ t \ \epsilon \mathbb{R}$

We have an elliptic equation with real coefficients. To solve the Generalized Dirichlet Problem one applies the Lax-Milgram lemma. However the Lax-Milgram lemma requires the bilinear form B to be coercive, which can be shown using the Garding inequality. The right condition for obtaining a unique solution turns out to be: $d \ + \ t < \ 0$. This condition insures that we are avoiding all the eigenvalues. Therefore we can define the Green's operator $G_t \ = \ L_t^{-1}$ on the space $\overset{\circ}{H}_1 (\Omega) \bigcap \ H_2(\Omega)$.

It can be shown that G_t is compact, and hence the Hilbert Schmidt theory [72] is available.

Case 2: $z \epsilon \mathbb{C}$

$$L_z(u) \ = \ a \, \frac{\partial^2 u}{\partial x^2} \ + \ b \, \frac{\partial^2 u}{\partial y^2} \ + \ du \ + \ zu \ = \ f \tag{2.85}$$

Let $t \ \epsilon \ \mathbb{R}$ be such that $d \ + \ t \ < \ 0$ (i.e. the Real Dirichlet Problem has a solution), then:

$$L_z \ (u) \ = \ \left(a \, \frac{\partial^2 u}{\partial x^2} \ + \ b \, \frac{\partial^2 u}{\partial y^2} \ + \ (d \ + \ t)u \right) \ + \ (z - t)u \ = \ f \tag{2.86}$$

by operating the Green's operator G_t on both sides we get:

$$u \ + \ (z - t)G_t u \ = \ G_t f \tag{2.87}$$

(and convesely this implies $L_z(u) = f$).

Corollary 2.4:

If $I_m(z) \neq 0$ then the Dirichlet problem $\begin{cases} L_z\,(u) = f \\ u/_{\partial\Omega} = 0 \end{cases}$ has a unique solution.

Remark:

We don't go into Regularity theory here, which enables one to conclude that a weak solution is a classical solution, and also investigate the boundary behavior of the solution. We refer to [31] for further details in this direction.

Relationship with the Green's function:

We have defined a Green's operator $G_t = L_t^{-1}$ in an appropriate Hilbert space. We now see how this concept relates to that of a Green's function which is defined as a solution to the following Direchlet Problem:

$$\begin{cases} L_z\,(G(x,\,\xi)) = \delta(x - \xi) \\ G|_{\partial\Omega} = 0 \end{cases} \tag{2.88}$$

It is easy to see that $\int_\Omega G(x,\,\epsilon)\; f(\epsilon)d\epsilon$ is a solution to the original Dirichlet problem, and by uniqueness:

$$(Gf)(x) = \int_\Omega G(x,\,\xi)\; f(\xi)d\xi \tag{2.89}$$

Thus the Green's function is the kernel of the Green operator.

Remark:

One can rigorously show that G is continuous but not differentiable (here we have a second order operator and the solution G is 2-degrees smoother than the the right hand side). This is another way to prove the campactness of the operator G.

2.2.2 Series Expansion of the Green's Function:

In this section we compute the Green's function for the following Boundary Value Problem.

$$\begin{cases} a_\beta \frac{\partial^2 H}{\partial \beta^2} + a_\alpha \frac{\partial^2 H}{\partial \alpha^2} + (\gamma^2 + w^2)H = -a_\beta \nu \quad \text{in} \quad \mathring{\Omega} \\ H|_{\partial\Omega} = 0 \end{cases} \tag{2.90}$$

To do so, we fix the α variable and look for a solution in terms of the eigenfunctions of a Sturm-Liouville problem in the α-direction. This leads to another one-dimensional Sturm-Liouville problem that we solve to get the final answer. This idea has been exploited (See [74]) to give an expression of the Green's function of the Laplace operator as an exponentially and uniformly convergent single sum.

Consider the operator:

$$\begin{cases} L_\alpha u = -a_\alpha \frac{\partial^2 u}{\partial \alpha^2} - (\gamma^2 + w^2)u \\ u(0) = u(\ell) = 0 \end{cases} \tag{2.91}$$

Then our equation can be rewritten as follows:

$$\begin{cases} a_\beta \frac{\partial^2 H}{\partial \beta^2} - L_\alpha H = -a_\beta \nu \\ H|_{\partial\Omega} = 0 \end{cases} \tag{2.92}$$

Thus the Green's function satisfies the following PDE:

$$\begin{cases} a_\beta \frac{\partial^2 K}{\partial \beta^2} - L_\alpha K = -\delta(\alpha - \zeta)\delta(\beta - \eta) \\ K_{\alpha,\beta} = 0 \text{ on the boundary of } [0,\ell] \times [0,\hat{\ell}] \end{cases} \tag{2.93}$$

Let $\varphi_k(\alpha)$ be the eigenfunctions of the operator $L_\alpha : k = 1, 2, \ldots$ We seek the Green's function in the following form:

$$K(\alpha, \beta, \zeta, \eta) = \sum_{k=1}^{\infty} a_k(\beta) \cdot \varphi_k(\alpha) \tag{2.94}$$

If we define the following weighted inner product:

$$\langle \varphi, \Psi \rangle = \iint_{\Omega} a_\beta \varphi \Psi \qquad \text{where } \Omega = [0, \ell] \times [0, \hat{\ell}] \tag{2.95}$$

Substituting the expression (2.94) into (2.93), using the orthogonality of the eigenfunctions, and the property of the Dirac function we get:

$$\begin{cases} a_k'' - \lambda_k a_k(\beta) = -\varphi_k(\zeta)\delta(\beta - \eta) \\ a_k(0) = -a_k(\hat{\ell}) = 0 \end{cases} \tag{2.96}$$

and this is a classic Sturm-Liouville problem.

The eigenvalue λ_n is given by:

$$\lambda_n = \frac{a_\alpha}{a_\beta} \cdot \frac{n^2\pi^2}{\ell^2} - \frac{\gamma^2 + w^2}{a_\beta} \qquad n = 1, 2, \dots \tag{2.97}$$

We therefore have to study three cases: $\lambda > 0, \lambda = 0, \lambda < 0$

Remark:

For each fixed ω there is only a finite number of negative eigenvalues. To solve the above we need only solve for:

$$\begin{cases} a'' - \lambda a = -\delta(\beta - \eta) \\ a(0) = a(\hat{\ell}) = 0 \end{cases} \tag{2.98}$$

where we have dropped the indices for convenience, and also the term $\varphi_n(\zeta)$ which will be accounted for by a multiplication of the resulting Green's function by the same factor.

Case 1: $\lambda = \mu^2 > 0$

Then it is easy to show that the Green's function of Problem (2.98) is given by:

$$G(\beta, \eta) = \frac{1}{\mu sh(\mu\hat{\ell})} \begin{cases} sh(\mu(\hat{\ell} - \eta)) \cdot sh[\mu\ \beta] & 0 < \beta < \eta \\ sh[\mu\eta] \cdot sh[\mu(\hat{\ell} - \beta)] & \eta < \beta < \hat{\ell} \end{cases} \tag{2.99}$$

To get the desired Green's function of Problem (2.96) we just multiply the above by:

$$\varphi_n(\zeta) = \sqrt{\frac{2}{\ell}} \sin\left(\frac{n\pi}{\ell}\xi\right) \tag{2.100}$$

to get:

$$G_n(\beta,\eta) = \sqrt{\frac{2}{\ell}} \cdot \frac{\sin\left(\frac{n\pi}{\ell}\zeta\right)}{\mu_n sh(\mu_n \hat{\ell})} \begin{cases} sh[\mu_n(\hat{\ell}-\eta)] \cdot sh[\mu_n \beta] & 0 < \beta < \eta \\ sh[\mu_n\ \eta] \cdot sh[\mu_n(\hat{\ell}-\beta)] & \eta < \beta < \hat{\ell} \end{cases} \tag{2.101}$$

where $\mu_n = \sqrt{\lambda_n}$

Case 2: $\lambda = -\mu^2 < 0$

In this case the Green's function for (2.98) is:

$$G_n(\beta,\eta) = \sqrt{\frac{2}{\ell}} \cdot \frac{\sin(\frac{n\pi}{\ell}\zeta)}{\mu_n \sin \mu_n \hat{\ell}} \begin{cases} \sin(\mu_n(\hat{\ell}-\eta)) \cdot \sin(\mu_n\beta) & 0 < \beta < \eta \\ \sin(\mu_n\eta) \cdot \sin(\mu_n(\hat{\ell}-\beta) & \eta < \beta < \hat{\ell} \end{cases} \tag{2.102}$$

Case 3: $\lambda = 0$

The Green's function for Problem (2.98) is in this case:

$$G_n(\beta,\eta) = \sqrt{\frac{2}{\ell}} \frac{\sin(\frac{n\pi}{\ell}\zeta)}{\hat{\ell}} \begin{cases} \beta(\hat{\ell}-\eta) & 0 < \beta < \eta \\ \eta(\hat{\ell}-\beta) & \eta < \beta < \ell \end{cases} \tag{2.103}$$

Remarks:

I. We notice that both (2.101) and (2.102) can be given by the same formula where μ_n is taken to be a complex number (either $i|\mu_n|$ or $|\mu_n|$). Also (2.103) can be obtained as the limit of either (2.101) of (2.102) (or the complex form of both) when $\mu_n \longrightarrow 0$. But it is much better to split the cases to have an efficient algorithm on the computer.

II. The procedure we have followed here presents two advantages over the classical expansion of a Green's function in terms of the eigenfunctions of the whole problem:

(1) It reduces the expression of the Green's function to a single sum indexed by n (instead of $\sum_n \sum_m$).

(2) The main advantage is the convergence property of this series:

   (a) there are only a finite number of negative eigenvalues.

   (b) Thus the main part of the series is given by (2.101) which can be expressed in terms of exponentials and one can show easily that this series is convergent (exponentially). The Green's function is given next.

Green's Function

$$
\begin{aligned}
K(\alpha,\beta,\zeta,\eta) &= \sum_{n=1}^{[n_0]} \frac{2}{\ell} \frac{\sin(\frac{n\pi}{\ell})\sin(\frac{n\pi}{\ell}\alpha)}{\mu_n \sin \mu_n \hat{\ell}} \begin{cases} \sin(\mu_n(\hat{\ell}-\eta))\cdot\sin(\mu_n\beta) \\ \sin(\mu_n\ \eta)\cdot\sin(\mu_n(\hat{\ell}-\beta)) \end{cases} \\
&+ e[n_0]\frac{2}{\ell}\frac{\sin(\frac{n\pi}{\ell}\zeta)\sin(\frac{n\pi}{\ell}\alpha)}{\hat{\ell}} \begin{cases} \beta(\hat{\ell}-\eta) \\ \eta(\hat{\ell}-\beta) \end{cases} \\
&+ \sum_{n=[n_0]+1}^{\infty} \frac{2}{\ell}\frac{\sin(\frac{n\pi}{\ell}\zeta)\sin(\frac{n\pi}{\ell}\alpha)}{\mu_n sh(\mu_n\hat{\ell})} \begin{cases} sh(\mu_n(\hat{\ell}-\eta))\cdot sh(\mu_n\beta) & 0<\beta<\eta \\ sh(\mu_n\eta)\cdot sh(\mu_n(\hat{\ell}-\beta)) & \eta<\beta<\hat{\ell} \end{cases}
\end{aligned}
\tag{2.104}
$$

whereby:

I. n_0 is defined by: $n_0 = \sqrt{\frac{\gamma^2+w^2}{a_2\pi^2}}$; and $[n_0]$ is the integer strictly less than n_0 (if n_0 is integer then $[n_0] = n_0 - 1$); $e[n_0] = \begin{cases} 0 & n_0 \text{ not integer} \\ 1 & n_0 \text{ integer} \end{cases}$

II. $\mu_n = \sqrt{|\lambda_n|} = \sqrt{\left|\frac{a_\alpha}{a_\beta}\frac{n^2\pi^2}{\ell^2} - \frac{\gamma^2+w^2}{a_\beta}\right|} n = 1, 2, \ldots$ Summing up, we have the following.

Theorem 2.5:

Under the assumptions of Section 2.2.2, the Green's function for the Boundary Value Problem (2.90) with real coefficients exists and is given by the expression:

$$G(x,y,x',y'/t) = G_-(x,y,x',y',t)+G_0(x,y,z',y',t)+G_+(x,y,x',y',t) \quad (2.105)$$

Where $G_\pm(x,y,x',y',t)$ corresponds to the negative (positive) eigenvalue of the Sturm-Liouville Problem, $G_0(x,y,x',y',t)$ corresponds to the zero eigenvalue.

They are given by the exponentially uniformly convergent series (2.104).

2.2.3 First Order Discretization of the Green's Function

The solution to the elliptic equation, governing the "motion" of the membrane in the frequency domain, is given by an integral operator whose kernel is the Green's function. Since it is difficult to compute multiple integrals on the computer we assume that the control is constant over small squares in the space domain and the solution to our PDE will be given by a double sum. Furthermore we can store these values of integrated Green's function over small squares to have a fast algorithm.

We have already shown that the Green's function, for $0 < \beta < \eta$, is given by:

$$\begin{aligned} K(\alpha,\beta,\zeta,\eta,\omega) = & \sum_{n=1}^{[n_0]-e[n_0]} \frac{2}{\ell} \frac{\sin(\frac{n\pi}{\ell}\zeta)\sin(\frac{n\pi}{\ell}\alpha)}{\mu_n \sin(\mu_n \hat{\ell})} \cdot \sin(\mu_n(\hat{\ell}-\eta))\sin(\mu_n\beta) \\ & + e[n_0]\frac{2}{\ell} \cdot \frac{\sin(\frac{n_0\pi}{\ell}\zeta)\sin(\frac{n_0\pi}{\ell}\alpha)}{\hat{\ell}} \cdot \beta(\hat{\ell}-\eta) \\ & + \sum_{n=[n_0]+1}^{\infty} \frac{2}{\ell} \cdot \frac{\sin(\frac{n\pi}{\ell}\zeta) \cdot \sin(\frac{n\pi}{\ell}\alpha)}{\mu_n sh(\mu_n \hat{\ell})} \cdot sh(\mu(\hat{\ell})) \cdot sh(\mu_n\beta) (2.106) \end{aligned}$$

and by symmetry:

$$K(\alpha,\beta,\zeta,\eta,\omega) = K(\zeta,\eta,\alpha,\beta,\omega) \qquad \text{for } \eta < \beta < \hat{\ell} \qquad (2.107)$$

The transfer function is thus given by:

$$H(\alpha,\beta,\omega) = \iint\limits_{\Omega} -a_\beta K_\omega(\alpha,\beta,\zeta,\eta)\nu(\zeta,\eta,\omega)d\zeta d\eta =< K,\nu >_{L^2(\Omega)} \qquad (2.108)$$

Now let us consider the following subdivision of Ω:

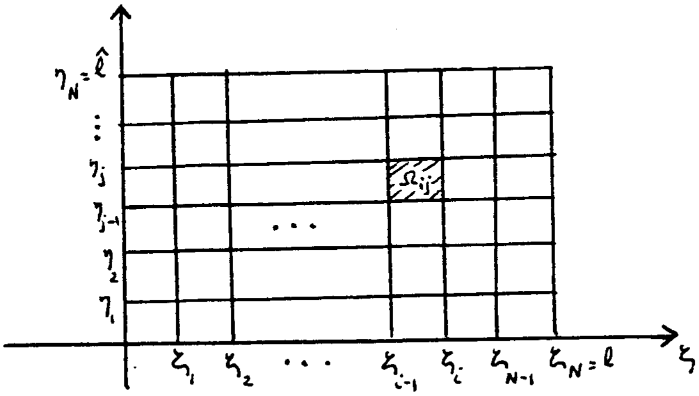


Fig. 3: Subdivision of the Membrane

Assume that the forcing term ν is constant over $\Omega_{ij} = [\zeta_i,\zeta_{i+1}] \times [\eta_j,\eta_{j+1}], i,j = 1,N$.

$$\nu(\zeta,\eta,\omega) = v_{ij}(\omega) \text{ for } (\zeta,\eta) \in \Omega_{ij} \qquad (2.109)$$

Then:

$$H(\alpha,\beta,\omega) = \sum_{i=1}^{N}\sum_{j=1}^{N} -a_\beta v_{ij}(\omega) \iint\limits_{\Omega_{ij}} K_\omega(\alpha,\beta,\zeta,\eta)d\zeta d\eta \qquad (2.110)$$

Thus we need to compute and store the integrals $\iint\limits_{\Omega_{ij}} K \; i,j = 1,N$ for a discrete number of values of $\omega_i \in (-\Omega_0,\Omega_0)$ and use interpolation techniques for other values of the frequency ω (Ω_0 is the estimated spectrum). To find expressions of the above integrals we need to distinguish three cases:

$$A: \quad \beta < \eta_j$$

$$B: \quad \beta > \eta_{j+1}$$

$$C: \quad \beta \in [\eta_j,\eta_{j+1}]$$

Case A: $\beta < \eta_j$

$$\iint\limits_{\Omega_{ij}} K_\omega(\alpha,\beta,\zeta,\eta)d\zeta d\eta = \underbrace{\iint\limits_{\Omega_{ij}} \sum_{n=1}^{[n_0]-e} \frac{2}{\ell} \cdot \frac{\sin(\frac{n\pi}{\ell}\zeta)\sin(\frac{n\pi}{\ell}\alpha)}{\mu_n \sin(\mu_n \hat{\ell})} \cdot \sin(\mu_n(\hat{\ell}-\eta))\sin(\mu_n\beta)d\zeta d\eta}_{I}$$

$$+ \underbrace{\iint\limits_{\Omega_{ij}} e \cdot \frac{2}{\ell} \frac{\sin(\frac{n_0^{\pi}}{\ell} \cdot \sin(\frac{n_0\pi}{\ell}\alpha)}{\hat{\ell}} \cdot \beta(\hat{\ell}-\eta)d\zeta dy}_{II} \tag{2.111}$$

$$+ \underbrace{\iint\limits_{\Omega_{ij}} \sum_{n=[n_0]-1}^{\infty} \frac{2}{\ell} \frac{\sin(\frac{n\pi}{\ell})\sin(\frac{n\pi}{\ell}\alpha)}{\mu_n sh(\mu_n\hat{\ell})} sh(\mu_n(\hat{\ell}-\eta)) \cdot sh(\mu_n\beta)d\zeta d\eta}_{III}$$

Computation of Integral I

$$I = \sum_{n=1}^{[n_0]} \frac{2}{\ell} \frac{\sin(\frac{n\pi}{\ell}\alpha)}{\mu_n \sin(\mu_n\hat{\ell})} \cdot \sin(\mu_n\ \beta) \underbrace{\iint\limits_{\Omega_{ij}} \sin(\frac{n\pi}{\ell}\zeta)\sin(\mu_n(\hat{\ell}-\eta))d\zeta d\eta}_{\mathcal{J}_I} \tag{2.112}$$

$$J_I = \int_{\zeta_i}^{\zeta_{i+1}} \sin(\frac{n\pi}{\ell})d\zeta \cdot \int_{\eta_j}^{\eta_j+1} \sin(\mu_n(\hat{\ell}-\eta))d\eta \tag{2.113}$$

Therefore by elementary computations we get:

$$J_I = \frac{\ell}{n\pi\mu_n} \left[\cos(\frac{n\pi}{\ell}\zeta_i) - \cos(\frac{n\pi}{\ell}\zeta_{i+1})\right] \left[\cos(\mu_n(\hat{\ell}-\eta_{j+1})) - \cos(\mu_n(\hat{\ell}-\eta_j))\right] \tag{2.114}$$

Hence:

$$I = \sum_{n=1}^{[n_0]-e} \frac{2}{n\pi\mu_n^2 \sin(\mu_n\hat{\ell})} \sin(\frac{n\pi}{\ell}\alpha) \cdot \sin(\mu_n\ \beta) \cdot \left[\cos(\frac{n\pi}{\ell}\zeta_i) - \cos(\frac{n\pi}{\ell}\zeta_{i+1})\right]$$

$$\cdot \left[\cos(\mu_n(\hat{\ell}-\eta_{j+1})) - \cos(\mu_n(\hat{\ell}-\eta_j))\right] \tag{2.115}$$

Computation of Integral II

Similarly integral II is given by:

$$II = \frac{2e}{n_0 \hat{\ell} \pi} \cdot \sin(\frac{n_0 \pi}{\ell} \alpha) \cdot \beta \cdot \left[\cos(\frac{n_0 \pi}{\ell} \zeta_i) - \cos(\frac{n_0 \pi}{\ell} \zeta_{i+1})\right]$$

$$\left[\hat{\ell}(\eta_{j+1} - \eta_j) - \frac{\eta_{j+1}^2 - \eta_j^2}{2}\right] \qquad (2.116)$$

Computation of integral III

Finally integral III is given by:

$$III = \sum_{n=[n_0]+1}^{\infty} \frac{2}{n\pi \mu_n^2 sh(\mu_n \hat{\ell})} \cdot \sin(\frac{n\pi}{\ell}\alpha) \cdot sh(\mu_n \beta) \left[\cos(\frac{n\pi}{\ell}\zeta_i) - \cos(\frac{n\pi}{\ell}\zeta_{i+1})\right]$$

$$\left[ch(\mu_n(\hat{\ell} - \eta_j)) - ch(\mu_n(\hat{\ell} - \eta_{j+1}))\right] \qquad (2.117)$$

The final result is that for $\beta < \eta_j$:

$$\iint_{\Omega_{ij}} K_\omega(\alpha, \beta, \zeta, \eta) d\zeta d\eta \triangleq S_1(\alpha, \beta, \zeta, \eta, \omega)$$

$$= \sum_{n=1}^{[n_0]-e} \frac{2}{n\pi \mu_n^2 \sin(\mu_n \hat{\ell})} \cdot \sin\left(\frac{n\pi}{\ell}\alpha\right) \cdot \sin(\mu_n \ \beta)$$

$$\cdot \left[\cos\left(\frac{n\pi}{\ell}\zeta_i\right) - \cos\left(\frac{n\pi}{\ell}\zeta_{i+1}\right)\right] \left[\cos(\mu_n(\hat{\ell} - \eta_{j+1})) - \cos(\mu_n(\hat{\ell} - \eta_j))\right]$$

$$+ \frac{2e}{n_0 \pi \hat{\ell}} \cdot \sin(\frac{n_0 \pi}{\ell}\alpha) \cdot \beta \cdot \left[\cos(\frac{n_0 \pi}{\ell}\zeta_i) - \cos(\frac{n_0 \pi}{\ell}\zeta_{i+1})\right] \left[\hat{\ell}(\eta_{j+1} - \eta_j) - \frac{\eta_{j+1}^2 - \eta_j^2}{2}\right]$$

$$+ \sum_{n=[n_0]+1}^{\infty} \frac{2}{n\pi \mu_n^2 sh(\mu_n \hat{\ell})} \cdot \sin(\frac{n_0 \pi}{\ell}\alpha) \cdot sh(\mu_n \ \beta) \left[\cos(\frac{n_0 \pi}{\ell}\zeta_i) - \cos(\frac{n_0 \pi}{\ell}\zeta_{i+1})\right]$$

$$\left[ch(\mu_n(\hat{\ell} - \eta_j)) - ch(\mu_n(\hat{\ell} - \eta_{j+1}))\right] \qquad (2.118)$$

Case B: $\beta > \eta_{j+1}$

By similar arguments one can get the following expression:

$$\iint\limits_{\Omega_{ij}} K_\omega(\alpha,\beta,\zeta,\eta)d\zeta d\eta \triangleq S_2(\alpha,\beta,\zeta,\eta,\omega)$$

$$= \sum_{n=1}^{[n_0]-e} \frac{2}{n\pi\mu_n^2 \cdot \sin(\mu_n\ \hat{\ell})} \cdot \sin(\frac{n_0\pi}{\ell}\alpha) \cdot \sin(\mu_n(\hat{\ell}-\beta)) \left[\cos(\frac{n_0\pi}{\ell}\zeta_i) - \cos(\frac{n_0\pi}{\ell}\zeta_{i+1})\right]$$

$$[\cos(\mu_n\ \eta_j) - \cos(\mu_n\ \eta_{j+1})]$$

$$+ \frac{2e}{n_0\pi\hat{\ell}} \cdot \sin(\frac{n_0\pi}{\ell}\alpha) \cdot \left[\cos(\frac{n_0\pi}{\ell}\zeta_i) - \cos((\frac{n_0\pi}{\ell}\zeta_{i+1})\right] \left[\frac{n_{j+1}^2 - \eta_j^2}{2}\right]$$

$$+ \sum_{n=[n_0]+1}^{\infty} \frac{2}{n\pi\mu_n^2 \cdot sh(\mu_n\hat{\ell})} \cdot \sin(\frac{n_0\pi}{\ell}\alpha) \cdot sh(\mu_n(\hat{\ell}-\beta)) \left[\cos(\frac{n_0\pi}{\ell}\zeta_i) - \cos((\frac{n_0\pi}{\ell}\zeta_{i+1})\right]$$

$$[ch(\mu_n\ \eta_{j+1}) - ch(\mu_n\ \eta_j)] \tag{2.119}$$

Case C: $\eta_j \le \beta \le \eta_{j+1}$

$$\iint\limits_{\Omega_{ij}} K_\omega(\alpha,\beta,\zeta,\eta)d\zeta d\eta \triangleq S_3(\alpha,\beta,\zeta,\eta,\omega)$$

$$= \underbrace{\int_{\zeta_i}^{\zeta_{i+1}} \int_{\eta_j}^{\beta} K_\omega(\alpha,\beta,\zeta,\eta)d\zeta d\eta}_{I} + \underbrace{\int_{\zeta_i}^{\zeta_{i+1}} \int_{\beta}^{\eta_{j+1}} K_\omega(\alpha,\beta,\zeta,\eta)d\zeta d\eta}_{II} \tag{2.120}$$

To get the value of Intergral I it is enough to plug in $\eta_{j+1} = \beta$ in Case B.

To get the value of Integral II it is enough to plug in $\eta_j = \beta$ in Case A.

Therefore:

$$I = \sum_{n=1}^{[n_0]-e} \frac{2}{n\pi\mu_n^2 \cdot \sin(\mu_n\ \hat{\ell})} \cdot \sin(\frac{n\pi}{\ell}\alpha) \cdot \sin(\mu_n(\hat{\ell}-\beta))$$

$$\left[\cos(\frac{n\pi}{\ell}\zeta_i) - \cos(\frac{n\pi}{\ell}\zeta_{i+1})\right]$$

$$[\cos(\mu_n\ \eta_j) - \cos(\mu_n\ \beta)] + \frac{2e}{n_0\pi\hat{\ell}}\cdot\sin(\frac{n_0\pi}{\ell}\alpha)$$

$$\cdot(\hat{\ell}-\beta)\cdot\left[\cos(\frac{n_0\pi}{\ell}\zeta_i) - \cos((\frac{n_0\pi}{\ell}\zeta_{i+1})\right]\left[\frac{\beta^2-\eta_j^2}{2}\right]$$

$$+\sum_{n=[n_0]+1}^{\infty}\frac{2}{n\pi\mu_n^2\cdot sh(\mu_n\hat{\ell})}\cdot\sin(\frac{n\pi}{\ell}\alpha)\cdot sh(\mu_n(\hat{\ell}-\beta))$$

$$\left[\cos(\frac{n\pi}{\ell}\zeta_i) - \cos((\frac{n\pi}{\ell}\zeta_{i+1})\right][ch(\mu_n\ \beta) - ch(\mu_n\ \eta_j)] \qquad (2.121)$$

$$II = \sum_{n=1}^{[n_0]-e}\frac{2}{n\pi\mu_n^2\cdot\sin(\mu_n\ \hat{\ell})}\cdot\sin(\frac{n\pi}{\ell}\alpha)\cdot\sin(\mu_n\ \beta))\cdot$$

$$\left[\cos(\frac{n\pi}{\ell}\zeta_i) - \cos(\frac{n\pi}{\ell}\zeta_{i+1})\right]\left[\cos(\mu_n(\hat{\ell}-\ \eta_{j+1})) - \cos(\mu_n(\hat{\ell}-\beta))\right]$$

$$+\frac{2e}{n_0\pi\hat{\ell}}\cdot\sin(\frac{n_0\pi}{\ell}\alpha)\cdot\beta\cdot$$

$$\left[\cos(\frac{n_0\pi}{\ell}\zeta_i) - \cos(\frac{n_0\pi}{\ell}\zeta_{i+1})\right]\left[\hat{\ell}(\eta_{j+1}-\beta) - \frac{\eta_{j+1}^2-\beta^2}{2}\right]$$

$$+\sum_{n=[n_0]+1}^{\infty}\frac{2}{n\pi\mu_n^2 sh(\mu_n\ \hat{\ell})}\cdot\sin(\frac{n\pi}{\ell}\alpha)\cdot sh(\mu_n(\hat{\ell}-\beta))$$

$$\left[\cos(\frac{n\pi}{\ell}\zeta_i) - \cos((\frac{n\pi}{\ell}\zeta_{i+1})\right]\left[ch(\mu_n(\hat{\ell}-\beta)) - ch(\mu_n(\hat{\ell}-\eta_{j+1}))\right] \qquad (2.122)$$

Using the expression (2.121) and (2.122) for integrals I and II we get:

$$\iint\limits_{\Omega} K_\omega(\alpha,\beta,\zeta,\eta)d\zeta d\eta \triangleq S_3(\alpha,\beta,\zeta,\eta,\omega) = I + II \qquad (2.123)$$

Summing up the results of this section we have:

Theorem 2.6:

Since the Green's function is continuous on the compact set Ω, the three series $S_1(\alpha,\beta,\zeta,\eta,\omega), S_2(\alpha,\beta,\zeta,\eta,\omega), S_3(\alpha,\beta,\zeta,\eta,\omega)$ given by the expressions

(2.118), (2.119) & (2.23) are uniformly converging to $G(\alpha, \beta, \zeta, \eta, \omega)$ in the intervals $\beta < \eta_j, \eta_j < \beta < \eta_{j+1}, \beta > \eta_{j+1}$ respectively.

Remarks

By studying the behavior of the Green's function at the boundary $\partial\Omega$ one can infer more about the rate of convergence of the three series as the mesh size shrinks to 0.

2.2.4 Asymptotic Behavior of G

1. We have already shown that the rate of convergence of the series is exponential with respect to the indices m, n.
2. The expression of the Green's function does not seem to be symmetric in terms of x, y, x', y'. However one can repeat the procedure while intergrating the Sturm-Liouville problem with respect to y first. The symmetric expression for G would then be the average of the two expressions.

   In terms of asymptotic behaviour this allows to approximate G on Ω except on a small square around the singularity x', y' (See Fig. 4)

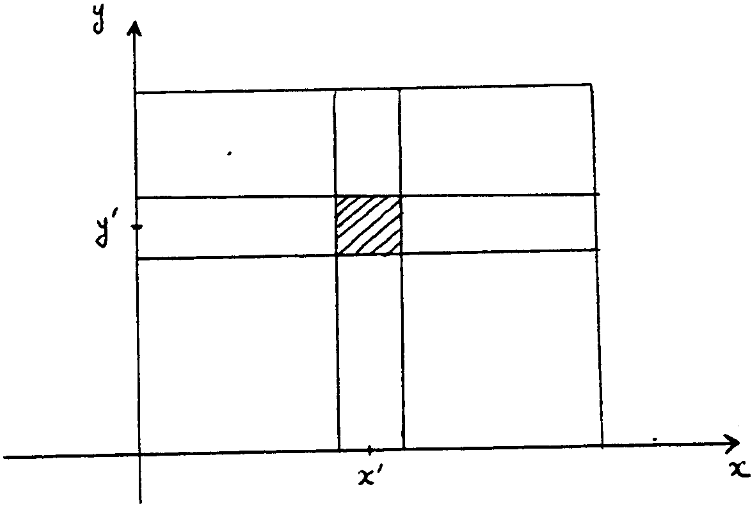


Fig. 4: Convergence in Ω\Shaded Square

3. In the shaded square around (x', y') one needs to compute an asymptotic expression for the terms of the series.

However these steps were not implemented because they introduce a huge computation of the formulas which are already very complicated (especially Step 2.). In fact the formulas given in the previous section are enough for our purposes, and the numerical simulation shows one can achieve a very high accuracy before there is an overflow (or underflow) on the computer.

2.2.5 Analytic Continuation of the "Real" Green's Function In the Case of Complex Parameter

Definition:

Let f be an analytic function in the domain $D \subset \mathbb{C}$ and g be an analytic function in the domain $\Omega \subset \mathbb{C}$ Assume that $f \equiv g$ on $D \cap \Omega$. We say that f is an analytic continuation of g to the domain D (and similarly g is the a.c. of f to Ω).

General Principle:

Suppose we are given a function $f(z)$ which is analytic in the domain Ω. Given $a \in \Omega$ we consider the largest disc centered at a where f is analytic (obviously the radius of this disc is the distance from a to the nearest singularity of f). We repeat this process for every point of Ω, in this way we get a new domain $D \supset \Omega$. This process can be repeated until one reaches a natural boundary. In general the analytic function obtained this way will be multivalued, we can however introduce a multi-sheeted Riemann surface where the function is single valued.

Riemann's Principle:

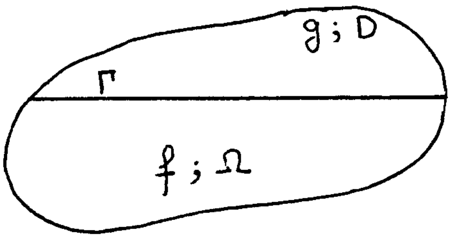


Fig. 5: Domain of Analyticity for Riemann's Principle

f is analytic in Ω and continuous on Γ. g is analytic in D and continuous on Γ.

Moreover $f = g$ on Γ

$$\text{Then } h = \begin{cases} f & \text{in } \Omega \\ f = g & \text{on } \Gamma \\ g & \text{in } D \end{cases} \quad \text{is \underline{analytic} on } \Omega \cup D \cup \Gamma.$$

Corollary 2.6: Schwarz Reflection Principle

Referring to Fig. 6 let $f(\lambda)$ be analytic on D real and continuous on the real axis. Then $f^*(\lambda^*)$ is analytic on D^*. This allows us to define an extension analytic on the whole domain $D \cup D^*$ as follows:

$$F = \begin{cases} f & \text{on } D \\ f = f^* & \text{on } D \cup D^* \\ f^* & \text{on } D^* \end{cases} \tag{2.124}$$

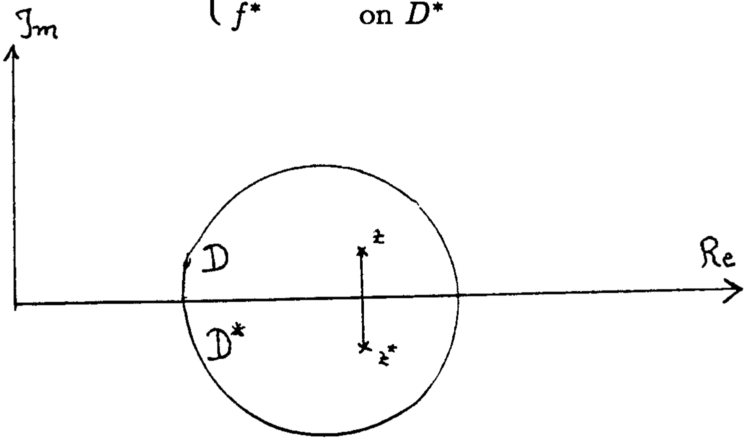


Fig. 6: Domain of Analyticity for Schwarz Reflection Principle

Remark:

We are going to use a refinement of the Schwarz Reflection principle:

Assume f is real analytic on a interval I of the real axis, then f can be analytically continued to the complex plane. To see this use the fact that the

Taylor coefficients of an analytic function determine that function uniquely. The values of f on the real axis allow us to compute its Tyalor coefficients. In the Schwarz reflection principle we merely ask for the function to be continuous on the real axis.

Using the previous refinement of the Schwarz reflection principle and regularity theory for elliptic PDE's [31] we get the following theorem:

Theorem 2.7:

The Real Green's function is analytic on the half-line not containing any eigenvalues of the Dirichlet problem, and the analytic continuation of the three series S_1, S_2, S_3 as given by (2.118), (2.119) (2.23) uniformly convergent to the Green's function with complex parameters.

2.2.6 Solution of the Elliptic System Using the Discretized Green's Function

Consider the following complex elliptic (PDE):

$$\begin{cases} a\frac{\partial^2 H}{\partial x^2} + b\frac{\partial^2 H}{\partial y^2} + pH = -bU \text{ in } \mathring{\Omega} \\ H|_{\partial\Omega} = 0 \\ \Omega = [0,\ell] \times [0,\hat{\ell}] \end{cases} \tag{2.125}$$

and a, b are as in equation (2.78). p is a complex parameter (depending on the frequency ω). Define the following quantities:

$$n_0 = \left(\frac{p\ell^2}{a\pi^2}\right)^{1/2} \tag{2.126}$$

$$e(n_0) = \begin{cases} 1 & n_0 \text{ integer} \\ 0 & \text{otherwise} \end{cases} \tag{2.127}$$

$$\mu_n = \left(\frac{a}{b}\frac{n^2\pi^2}{\ell^2} - \frac{p}{b}\right)^{1/2} \tag{2.128}$$

Assume that $\Omega = [0, \ell] \times [0, \hat{\ell}]$ is subdivided into small rectangles:

$$\Omega_{ij} = [x_i, x_{i+1}] \times [y_j, y_{j+1}] \begin{array}{l} i = 1, \ldots N+1 \\ j = 1, \ldots M+1 \end{array}$$

$$\text{Define} \qquad h = \frac{\ell}{N} \quad k = \frac{\hat{\ell}}{M} \tag{2.129}$$

Assume that the control is constant in space over the rectangle Ω_{ij} and defined by its value at the center

$$u(x, y, p) = u_{ij}(p) = U(\frac{x_i + x_{i+1}}{2}, \frac{y_j + y_{j+1}}{2}, p) \tag{2.130}$$

$$(x, y) \in \Omega_{ij}$$

Now, let $G_{ij}(x, y, p)$ be the average Green function over Ω_{ij}

$$G_{ij}(x, y, p) = \iint\limits_{\Omega_{ij}} K(x, y, \zeta, \eta, p) d\zeta d\eta. \tag{2.131}$$

The solution to the complex PDE will be given by:

$$H(x, y, p) = \sum_{i=1}^{N} \sum_{j=1}^{M} b u_{ij}(p) G_{ij}(x, y, p) \tag{2.132}$$

Let us make a slight change of notation which will prove useful in the programming of the algorithm:

$$G_{ij}(x, y, p) = G(x, x_i, x_{i+1}, y, y_j, y_{j+1}, p) \tag{2.133}$$

$$u_{ij}(p) = u(x_i, x_{i+1}, y_j, y_{j+1}, p) \tag{2.134}$$

The function $G_{ij}(x, y, p)$ will be given by the following:

<u>1. $y < y_j$:</u>

$$\begin{aligned} G^I_{ij}(x, y, p) = -\Big\{ & G_1(x, x_i, x_{i+1}, \hat{\ell} - y, \hat{\ell} - y_j, \hat{\ell} - y_{j+1}, p) \\ & + G_2(x, x_i, x_{i+1}, \hat{\ell} - y, \hat{\ell} - y_j, \hat{\ell} - y_{j+1}, p) \Big\} \end{aligned} \tag{2.135}$$

2. $y > y_{j+1}$:

$$G_{ij}^{II}(x,y,p) = G_1(x, x_i, x_{i+1}, y, y_j, y_{j+1}, p) + G_2(x, x_i, \ldots) \tag{2.136}$$

3. $y_j < y < y_{j+1}$:

$$\begin{aligned} G_{ij}^{III}(x,y,p) &= G^{II}(x, x_i, x_{i+1}, y, y_j, y, p) \\ &\quad + G^{I}(x, x_i, x_{i+1}, \hat{\ell} - y, \hat{\ell} - y_{j+1}, \hat{\ell} - y, p) \end{aligned} \tag{2.137}$$

$$\begin{aligned} G_1(\ldots) = &\left\{ \sum_{\substack{n=1 \\ n \neq n_0}}^{\infty} \frac{2}{n\pi\mu_n^2 sh(\mu_n \hat{\ell})} \sin(\frac{n\pi}{\ell}x) \cdot sh(\mu_n(\hat{\ell} - y)) \right\} \\ &\times \left\{ \cos(\frac{n\pi}{\ell}x_i) - \cos(\frac{n\pi}{\ell}x_{i+1}) \right\} \times \{ch(\mu_n y_{j+1} - ch(\mu_n y_j)\} \end{aligned} \tag{2.138}$$

$$\begin{aligned} G_2(\ldots) = &\, e(n_0) \cdot \frac{2}{n_0\pi\hat{\ell}} \cdot \sin(\frac{n\pi}{\ell}x) \cdot (\hat{\ell} - y) \cdot (\cos(\frac{n_0\pi}{\ell}x_i) - \cos(\frac{n_0\pi}{\ell}x_{i+1})) \\ &\times (\frac{y_{j+1}^2 - y_i^2}{2}) \end{aligned} \tag{2.139}$$

Theorem 2.8:

Consider the complex elliptic PDE:

$$\begin{cases} a\frac{\partial^2 H}{\partial x^2} + b\frac{\partial^2 H}{\partial y^2} + pH = -bU \text{ in } \overset{\circ}{\Omega} \\ H|_{\partial\Omega} = 0 \\ \Omega = [0,\ell] \times [0,\hat{\ell}] P \epsilon \mathbb{C} \end{cases} \tag{2.140}$$

Then

$$H_{MN}(x,y,p) = \sum_{i=1}^{N}\sum_{j=1}^{M} b u_{ij}(p) G_{ij}(x,y,p) \tag{2.141}$$

Converges uniformly to the solution $H(x,y,p)$ as $M,N \to \infty$, where $G_{ij}(x,y,p)$ is given by:

$$\begin{aligned} G_{ij}(x,y,p) &= G_{ij}^{I}(x,y,p) \quad \text{for } \ y < y_j \quad \text{(as in(2.135)} \\ &= G_{ij}^{II}(x,y,p) \quad \text{for } \ y > y_{j+1} \quad \text{(as in(2.136))} \end{aligned} \tag{2.142}$$

and $u_{ij}(x,y,p)$ are as in (2.134).

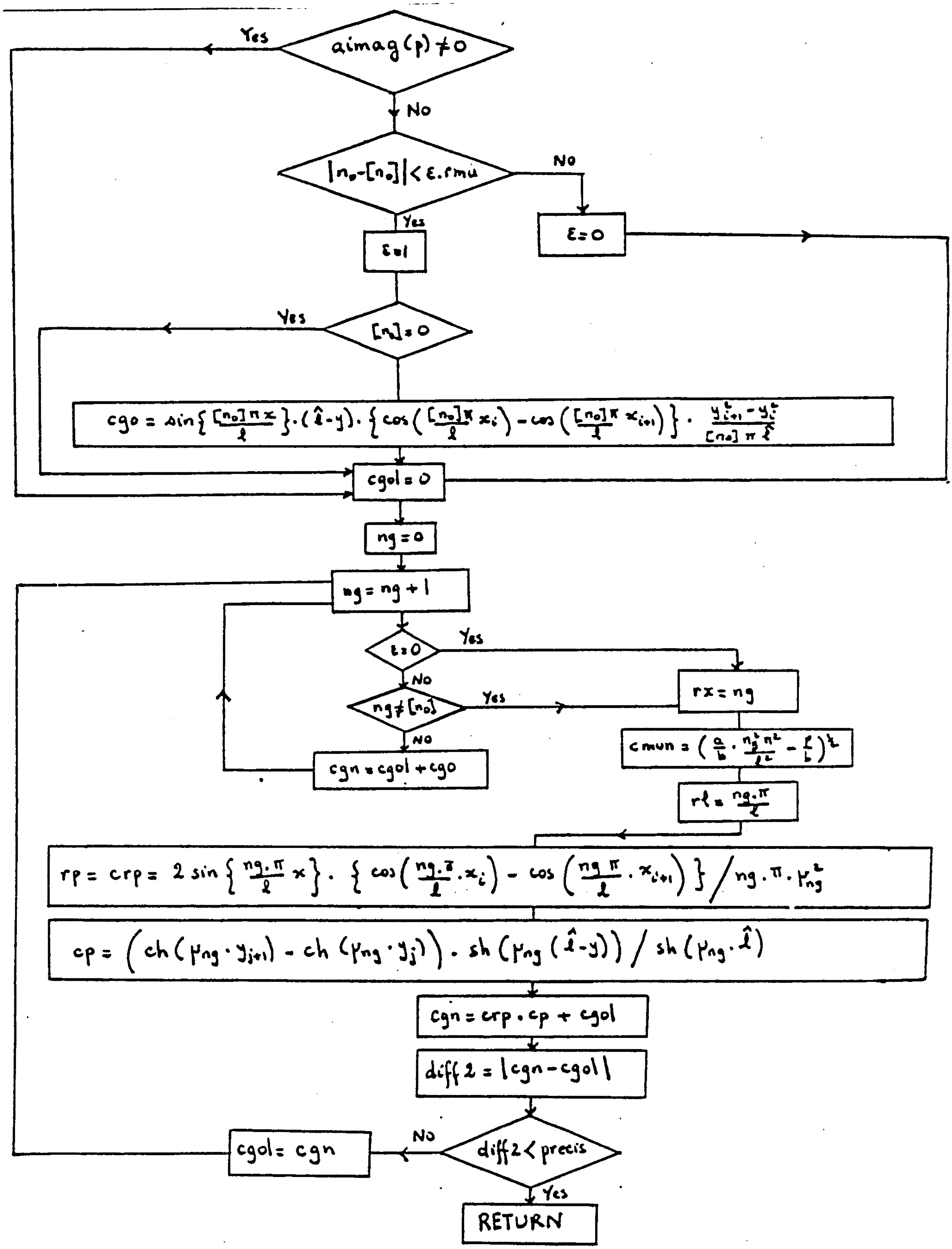

$aimag(p) \neq 0$
Yes
No
$|n_o - [n_o]| < \varepsilon . rmu$
No
Yes
$\varepsilon = 1$
$\varepsilon = 0$
Yes
$[n_o] = 0$
$cgo = sin\left\{\frac{[n_o]\pi x}{l}\right\} . (\hat{l} - y) . \left\{cos\left(\frac{[n_o]\pi}{l} x_i\right) - cos\left(\frac{[n_o]\pi}{l} x_{i+1}\right)\right\} . \frac{y_{i+1}^2 - y_i^2}{[n_o]\, \pi\, \hat{l}}$
$cgol = 0$
$ng = 0$
$ng = ng + 1$
$\varepsilon = 0$
Yes
No
$ng \neq [n_o]$
Yes
No
$cgn = cgol + cgo$
$rx = ng$
$cmun = \left(\frac{a}{b} . \frac{ng^2 \pi^2}{l^2} - \frac{f}{b}\right)^{\frac{1}{2}}$
$rl = \frac{ng . \pi}{l}$
$rp = crp = 2\, sin\left\{\frac{ng . \pi}{l} x\right\} . \left\{cos\left(\frac{ng . \pi}{l} . x_i\right) - cos\left(\frac{ng\, \pi}{l} . x_{i+1}\right)\right\} / ng . \pi . \mu_{ng}^2$
$cp = \left(ch(\mu_{ng} \cdot y_{j+1}) - ch(\mu_{ng} \cdot y_j)\right) . sh(\mu_{ng}(\hat{l} - y)) / sh(\mu_{ng} . \hat{l})$
$cgn = crp . cp + cgol$
$diff2 = |cgn - cgol|$
No
$diff2 < precis$
$cgol = cgn$
Yes
RETURN

Fig. 7: Organingramme for the Computation of Green's Function

2.2.7 Numerical Implementation and Applications

The reason why we need the theorem on the analytic continuation of the Green's function is that the elliptic PDE with complex parameter does not decouple into a system of two independent elliptic PDE's. The following example will clarify this point.

Consider the elliptic BVP:

$$\begin{cases} \frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + zU = f \\ U|_{\partial\Omega} = 0 \end{cases} \tag{2.143}$$

We take $z = j\alpha$, a pure-imaginary number. Now let $U = \nu + jw$. Then the above BVP becomes:

$$\begin{cases} \frac{\partial^2 \nu}{\partial x^2} + \frac{\partial^2 \nu}{\partial y^2} - \alpha w = f \\ \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \alpha \nu = 0 \end{cases} \tag{2.144}$$

Obviously this system cannot be decoupled. To obtain a closed form solution we take f to be an eigenfunction of the system:

$$f(x,y) = C\varphi_{mn}(x,y) \tag{2.145}$$

We look for solutions of the form:

$$\begin{cases} \nu(x,y) = A\varphi_{mn}(x,y) \\ w(x,y) = B\varphi_{mn}(x,y) \end{cases} \tag{2.146}$$

provided the compability conditions hold:

$$\begin{cases} C = \lambda_{mn}^2 + \alpha^2 \\ A = \lambda_{mn} \\ B = -\alpha \end{cases} \tag{2.147}$$

This example serves two purposes:

(1) it shows that the elliptic system cannot be decoupled in general, and we have to use theorem [2.8] on the analytic continuation of the solution to the elliptic PDE with real coefficients.

(2) We can compute the solution using the complexified Green's function and compare to the exact solution.

The numerical results are satisfactory, however, we report the numerical results only for the next example which is much more involved than the present one.

Consider the elliptic system, or equivalently, the complex elliptic partial differential equation:

$$\begin{cases} a\frac{\partial^2 f}{\partial x^2} + b\frac{\partial^2 f}{\partial y^2} + [(d+\omega^2) + j \cdot c\omega]f = u(x,y,\omega) \quad \text{in } \overset{\circ}{\Omega} \\ f|_{\partial\Omega} = 0 \end{cases} \tag{2.148}$$

Let

$$\begin{aligned} &\zeta_0 = 0 < \zeta_1 < \ldots < \zeta_{n+1} = L \\ &\eta_1 = 0 < \eta_1 < \ldots < \eta_{n+1} = \hat{L} \qquad \text{be a subdivision of } \Omega \end{aligned} \tag{2.149}$$

If $\Psi(x,y,z)$ is any function of $(x,y) \in \Omega$ with parameter z we adopt the notation: $\Psi_{mn}(z) = \Psi(x,y,z)$ for $(x,y) \in \Omega_{mn} = (\zeta_m, \zeta_{m+1}) \times (\eta_n, \eta_{n+1})$.

Then

$$f(x,y,\omega) = \sum_{n=1}^{N}\sum_{n=1}^{N} u(x_n, y_n, \omega) \cdot G_{mn}(x,y,\omega) \tag{2.150}$$

where (x_m, y_n) in the center of Ω_{mn}.

The above solution is an immediate application of the results of Theorem 2.8 where we take:

$$p = (d+\omega^2) + jc\omega \tag{2.151}$$

As an example we take:

$$u(x,y,\omega) = \left[(\lambda_{mn} + d + \omega^2) + jc\omega\right] \cdot [\alpha\frac{\cos\omega}{\omega^4+1} + \beta e^{-\omega} \cdot \sin\omega + \delta]\varphi_{mn}(x,y) \tag{2.152}$$

where $\varphi_{mn}(x,y)$ is the eigenfunction cooresponding to the eigenvalue λ_{mn}. The solution is therefore given by:

$$f(x,y,\omega) = (\frac{\alpha \cdot \omega}{\omega^4+1} + \beta e^{-\omega} \sin\omega + \delta) \sin\frac{m\pi}{L} x \cdot \sin\frac{n\pi}{\hat{L}} y \qquad (2.153)$$

Next we report the results of the numerical simulation of this example: We compare the exact solution given by equation (2.153) and compare it to the computed solution generated by the algorithm using equations (2.150).

The parameters of the simulation are as follows:

- $L = \hat{L} = 1$ are the length and width of the membrance.
- $m = n = 1$ defines the eigenfunction and corresponding eignevalue.
- $a = b = 1, c = -1, d = 1$ are the coefficients of the PDE.
- $\alpha = \beta = \delta = 1$ are the parameters in equations (2.152) and (2.153).
- The precision on the Green's function is: 10^{-7}
- The desired frequency is: $\omega = 0.0$
- We also need to specify which point (x,y) on the membrance to take: $x = y = 0.5$

Finally we need to provide the number of grid points N (as in Fig. 1). In the next table we report the results for two different values of N:

(1) N = 4: in this case there are 16 points on the membrance and the results are satisfactory.

(2) N = 100: there are 10^4 points on the membrance and this is an almost perfect case. We do these experiments to see what minimal number of grid points will still lead to a satisfactory precision. Later when we implement the transfer function, the optimal gains – we will have a huge memory requirement and we therefore need to have an idea about

the number of grid points to take in order to be able to implement the algorithm with the capacity requirements for a mini-computer.

Number of mesh points	**4**	**100**
Exact solution	2.0000	2.0000
Computed solution	1.8934	1.9998

Fig. 8: Comparison of the Exact Solution and the Computed Solution of the Elliptic System

2.3 Spectral Factorization

In this section we consider the space L^2 of square integrable functions over $\mathbb{R}$ and its subspace L^2_+ which consists of L^2 functions that have anti-casual inverse Fourier transform.

Given $f \in L^2$ we give a formula for computing its projection f^+ onto L^2_+ using Hilbert transform, section 2.3.1.

In section 2.3.2 we give the expressions of the boundary values of an analytic function in terms of the Hilbert transform of this function. Finally we combine the previous results to get a scalar spectral factorization theorem for functions in L^2.

In section 2.3.4 we present related numerical results.

2.3.1 Hilbert Transform and Projection onto L^+_2

Given $f \in L^2$ we define its Hilbert transform by:

$$Hf(t) = \frac{1}{\pi}\int_{-\infty}^{+\infty^*} \frac{f(\tau)}{t-\tau} d\tau (2.154)$$

* indicates the integral should be understood in the Cauchy principal value sense.

Let us also recall the definition of the Fourier transform of f:

$$F(s) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{isx}\ f(x)dx \tag{2.155}$$

Now consider

$$T \in \mathrm{L}^2 : T(s) = R(s) + i\mathcal{J}(s) \tag{2.156}$$

decomposed into real and imaginary parts. Then we have:

$$HT(\omega) = HR(\omega) + iH\mathcal{J}(\omega) \tag{2.157}$$

Let us compute the inverse Fourier Transform of HT:

$$\begin{aligned} \mathcal{F}^{-1}\{HT(\omega)\} &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-i\omega t} HT(\omega) d\omega \\ & \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-i\omega t} \frac{1}{\pi} \int_{-\infty}^{+\infty} \frac{T(s)}{\omega - s} ds d\omega \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-i\omega t} \frac{1}{\pi} \int_{-\infty}^{+\infty} \frac{R(s)}{\omega - s} ds d\omega \\ & + i[\ldots \mathcal{J}(s) \ldots] \end{aligned} \tag{2.158}$$

If we interchange the order of integration (using Tonnelli theorem) and use the fact that:

$$\frac{1}{\pi} \int_{-\infty}^{+\infty} \frac{e^{-i\omega t}}{-s} d\omega = -ie^{-ist} sgn(t) \tag{2.159}$$

we get the following important equation relating the inverse Fourier Transform of an L^2-function with the inverse Fourier Transform of its Hilbert Transform:

$$\mathcal{F}^{-1}\{HT\} = -i.sgn(t) \cdot \mathcal{F}^{-1}(T) \tag{2.160}$$

Now let us define:

$$T_{+} = \frac{1}{2}(T - iHT) \tag{2.161}$$

$$T_{-} = \frac{1}{2}(T + iHT) \tag{2.162}$$

By using equation (59) we get immediately:

$$\mathcal{F}^{-1}(T_+) = \begin{cases} 0 & t > 0 \\ \mathcal{F}^{-1}(T) & t < 0 \end{cases} \tag{2.163}$$

and

$$\mathcal{F}^{-1}(T_-) = \begin{cases} \mathcal{F}^{-1}(T) & t > 0 \\ 0 & t < 0 \end{cases} \tag{2.164}$$

i.e. T_+ (resp. T_-) is the projection of T onto the space L^2_+(resp. L^2_-), with the obvious definitions for L^2_+ and L^2_-. The above (60) and (61) give explicit formulas for the projections T^+ and T^- of the complex function T onto the spaces L_2^+ and L_2^- resp., in terms of its Hilbert transform.

2.3.2 Boundary Values of Analytic Functions and Hilbert Transform

Now we quote the following theorem which will play a key role in determining the relationship between the boundary behavior of an analytic function with its Hilbert Transform:

Theorem 2.9:

Let $\varphi \in L_2^+, a(z)(z = x + iy)$ is continuous for $y \geq 0$ and analytic for $y > 0$ except possibly for a pole of order n at $z = \zeta$. Moreover $a(z)$ is bounded, except possibly at $z = \zeta$ (i.e. $a(z)$ is bounded in the upper half-plane outside of every neighborhood of $z = \zeta$). Then for suitable $\alpha_1, \alpha_2, \ldots \alpha_n$:

$$a(x) - \varphi(x) - \sum_{k=1}^{n} \frac{\alpha_k}{(x-\zeta)^k} \in L_2^+ \tag{2.165}$$

(If $a(x)$ has no poles then $a(x)\varphi(x) \in L_2^+$). (For a proof see [35]).

The next theorem gives explicit formulas for the boundary values of an analytic function using Hilbert transform:

Theorem 2.10:

Let $\varphi(\tau) \in L_2(-\infty, +\infty)$ and consider

$$q(\tau) = \frac{1}{\pi i} \int_{-\infty}^{+\infty} \frac{\varphi(\tau) d\tau}{\tau - z} y = I_m(z) \neq 0 \tag{2.166}$$

Then $q(z)$ is a bounded analytic function for $y > 0$ and $y < 0$. Its boundary values are given by:

$$\lim_{y \to \pm 0} q(z) = q^{\pm}(z) = \pm\varphi(x) + iH\varphi(x) \tag{2.167}$$

The next Theorem 2.11 is really the one that give us the spectral factor we need:

Theorem 2.11:

Let $p(x)$ be such that:

$$\lim_{|x| \to \infty} p(x) = 1 \tag{2.168}$$

Define log such that: $\lim_{x \to \pm\infty} \log p(x) = 0$

Assume $\log p(x) \in L^2(-\infty, +\infty)$. Then there exists a bounded function $q(z)$ analytic for $Im(z) \neq 0$ such that:

$$p(x) = \frac{q^+(x)}{q^-(x)} \tag{2.169}$$

where $q^{\pm}(x) = \lim_{y \to \pm 0} q(z)$

Proof:

Let $\varphi(x) = \frac{1}{2} \log p(x) \in L^2(-\infty, +\infty)$ define $\log q(z) = \frac{1}{\pi_i} \int_{-\infty}^{+\infty} \frac{1/2 \log p(\tau)}{\tau - z} d\tau$

q is bounded analytic for $y \neq 0$

$$\lim_{y \to \pm 0} q(z) = q^{\pm} = \exp\{\pm\frac{1}{2} \log p(x) k + \frac{i}{\pi} \int_{-\infty}^{+\infty} \frac{\frac{1}{2} \log p(\tau)}{\tau - x} d\tau\}$$

$$q^{\pm} = \exp \frac{1}{2}(iH \log p \pm \log p)$$

Or $$\frac{q^+}{q^-} = \exp(\frac{1}{2} 2 \log p) = p. \tag{2.170}$$

2.3.3 Applications to Spectral Factorization

Now consider a function $f(x)$ satisfying the following constraints:

$$\lim_{|x|\to\infty} f(x) = 1 \tag{2.171}$$

$$\text{If } \Psi(x) = log f(x)) \text{ then } \Psi \epsilon L^2 \tag{2.172}$$

Then applying the previous results to the function

$$\Psi(x) = \log(f(x)) \tag{2.173}$$

(one has to be careful when defining which branch of log is using) We can show that $f(x)$ decomposes into

$$f(x) = f^+(x) \cdot f^-(x) \tag{2.174}$$

where f^+ and f^- are called the spectral factors (anti causal and causal factors respectively). For instance:

$$f^-(x) = \sqrt{f(x)} \exp\{-\frac{i}{2} H \log f(x)\} \tag{2.175}$$

We can therefore announce the scalar spectral factorization theorem:

Theorem 2.12:

Let $f(x)$ be such that:

$$\lim_{|x|\to\infty} f(x) = 1 \tag{2.176}$$

and assume that $log f(x) \epsilon L^2(-\infty, +\infty)$. The $f(x)$ admits a spectral factorization:

$$f(x) = f^+(x) \cdot f^-(x) \tag{2.177}$$

$f^-(x)$ (respectively $f^+(x)$) is the anti-casual (respectively casual) factor of the factorization and belongs to L^2_- (respectively L^2_+). It is given in terms of the Hilbert Transform as:

$$f^-(x) = \sqrt{f(x)} exp\left\{-\frac{i}{2} H \; log \; f(x)\right\} \tag{2.178}$$

Remarks:

1. When solving the linear regulator problem for the membrane problem, one shows that it is nothing but a minimum norm problem in an appropriate Hilbert space. The application of the Wiener-Hopf technique (which is just the projection method in this case) requires computing the anti-causal spectral factor (for stability purposes), and that's why we need $f^{-}(x)$.

2. For the multidimensional case we use Davis algorithm, [23] and compute the projection and the initial condition by the Hilbert transform technique (see Chapter [1]).

3. The numerical results are very satisfying and one can assign any desired precision (that can be achieved on a computer).

4. Here we have given the most direct mathematical approach to get the spectral factorization theorem which is enough for our purposes.

5. If we take a more modern (and more general as well) approach we can deduce our Theorem 2.12 as a corollary of the Abstract theory of Singular Operators (see [47, 49]) and the General Wiener-Hopf Factorization in Banach spaces (see [63]). For instance the projection formular in section 2.3.1 and the properties of the Hilbert Transform can be obtained by considering the symbol of H:

$$S(\omega) = FHF^{-1} = -isgn(\omega) \tag{2.179}$$

   Thus:

$$\mid S(\omega) \mid = 1 \Rightarrow \parallel Hf \parallel_2 = \parallel f \parallel_2 \qquad \text{(norm preserving)}$$

$$S^2(\omega) = -1 \Rightarrow \quad H^2 = -I \tag{2.180}$$

6. The condition $logf(x)\epsilon L^2(-\infty, +\infty)$ in our theorem is not a restriction since it already contains all cases of interest. However, its only reason of existence is to be able to apply the Hilbert Transform:

Y. V. Sokhotski investigated the behavior of the Cauchy type integral and derived similar formulae in 1873 (see [26]) for functions satisfying a Holder condition. I. Plemelj (1908) and later N.I Privalov (1918) gave a rigorous treatment (see [26]). Later on F. Riesz proved the Hilbert transform to be a continuous operator on L_p (see [47]) and this is in sharp contrast to the Fourier Transform.

7. In the references cited in remark 6 one can find the application of these theories to problems of Mathematical Physics (Elasticity, Hydrodynamics, Gas dynamics - - -). Further investigation is needed to show how to apply our algorithm to give a powerful method for computing the solution of Riemann BVP, Hilbert BVP - - -

2.3.4 Spectral Factorization and Control of One Dimensional Systems

We consider one-dimensional structures admitting one space variable such as strings, beams, etc. We also assume that we have one actuator (e.g. point control, boundary control, shape control, - - -) and N sensors (the arbitrary location of which is left to the designer). Actually it has been shown ([6] and references therein) that all one dimensional structures need only one actuator/sensor to achieve controllability/observability.

Let

$$G(j\omega) = CR(j\omega; A)B \tag{2.181}$$

be the $N \times 1$ transfer function of the structure (see section 2.4 and reference [22]). We can now state the following:

Theorem 2.13:

Consider a one-dimensional structure with transfer function $G(j\omega)$ as in (2.181).

Then

$$F(j\omega) = 1 + G^{*}G(j\omega) \tag{2.182}$$

is a scalar function and let $F^{-}(j\omega)$ be its anti-casual spectral factor given in terms of the Hilbert Transform as in (2.178). Then the optimal state feedback gain (see section 2.4) is given by:

$$[B^{*}K].x = \frac{1}{2\pi}\int_{-\infty}^{+\infty}[F^{-}(j\omega)]^{-1}G^{*}(j\omega)CR(j\omega)CR(j\omega;A)x\,d\omega \qquad (2.183)$$

Remarks:

1. This theorem settles the question of quadratic optimal control for one dimensional structures.

2. No restrictive assumptions concerning A,B,C are made and the theorem covers for instance boundary control, such as the one we successfully implemented for a beam with structural damping in Chapter 5.

5. This theorem covers many interesting flexible space structures such as NASA's SCOLE truss structure [5]

2.3.5 Application and Numerical Results

Next we present some examples to test our spectral factorization algorithm.

Example 1 (Hilbert transform of a window)

$$\varphi(x) = \begin{cases} 1 & |x| < 1 \\ 0 & \text{otherwise} \end{cases} \qquad (2.184)$$

$$H\varphi(x) = \frac{1}{\pi}\int_{-1}^{1^{*}}\frac{dy}{x-y} = \frac{1}{\pi}\log\left|\frac{x+1}{x-1}\right| \qquad (2.185)$$

Example 2 (spectral factorization of an exponential)

$$\text{Let} \qquad f(t) = e^{-\alpha|t|} \rightarrow F(j\omega) = \frac{\alpha}{\alpha^{2}+\omega^{2}}$$

$$f^{+}(t) = e^{-\alpha t}u(t) \rightarrow F^{+}(j\omega) = \frac{1}{\alpha + j\omega} \qquad (2.186)$$

Example 3:

$$f(s) = \frac{s^2+2}{s^2+1} \qquad (\text{notice} \lim_{|s|\to\infty} f(s) = 1)$$

$$f^+(s) = \frac{s+i\sqrt{2}}{s+i} \tag{2.187}$$

The reason we do not report the results for these "simple"examples is because we later present the numerical spectral factorization for the membrance, and also because we present next the numerical results showing the performance of the algorithm under the much severe conditions of singular perturbations. The algorithm performs well when the perturbation of the transfer function of the form $\frac{\varepsilon}{s-s_0}$ where ε is a small parameter. In the next table we present the numerical results obtained when we decrease the perturbation parameter and we compare the computed spectral factor with the limit spectral factor. It shows beyond any doubt that we have convergence, we get a precision of 10^{-4} that can be improved if we increase the number of points.

Perturbation Parameter	Computed Spectral Factor	Limit Spectral Factor
0.10	1.1301888227	1.0000000000
	1.3011419773	1.4142135382
0.05	1.0681126118	1.0000000000
	1.3607549667	1.4142135382
0.025	1.0347822905	1.0000000000
	1.3882908821	1.4142135382
0.0125	1.0175929070	1.0000000000
	1.4014406204	1.4142135382
0.00625	1.0088740587	1.0000000000
	1.4078550339	1.4142135382
0.003125	1.0044845343	1.0000000000
	1.4110213518	1.4142135382

Fig. 9: Performance of the Spectral Factorization Algorithm Near a Singularity in Terms of a Small Parameter.

Perturbation Parameter	Computed Spectral Factor	Limit Spectral Factor
0.0007812500	1.0011790991 1.4133784771	1.0000000000 1.4142135382
0.0000000954	1.0000748634 1.4141606092	1.0000000000 1.4142135382
0.0000000477	1.0000749826 1.4141603708	1.0000000000 1.4142135382
0.0000000238	1.0000748634 1.4141604900	1.0000000000 1.4142135382
0.0000000109	1.0000748634 1.4141604900	1.0000000000 1.4142135382
0.0000000060	1.0000748634 1.4141604900	1.0000000000 1.4142135382
0.0000000030	1.0000748634 1.4141604900	1.0000000000 1.4142135382
0.0000000015	1.0000748634 1.4141604900	1.0000000000 1.4142135382
0.0000000007	1.0000748634 1.4141604900	1.0000000000 1.4142135382
0.0000000004	1.0000748634 1.4141604900	1.0000000000 1.4142135382
0.0000000002	1.0000748634 1.4141604900	1.0000000000 1.4142135382

Fig. 9: (Cont'd): Performance of the Spectral Factorization Algorithm Near a Singularity in Terms of a Small Parameter.

2.4 Gain Computations

We first describe briefly the Wiener-Hopf approach to computing the optimal gains, following Davis [22].

Given a linear evolution equation:

$$\begin{cases} \frac{dx}{dt} = Ax + Bu \\ x(0) = x_0 \\ y = Cx \end{cases} \tag{2.188}$$

where A is an infinitesimal generator of a C_0-semi-group in a Hilbert space H

We want to solve the variational problem:

$$\min_u \int_0^\infty (\|x\|^2 + \|u\|^2) dt \tag{2.189}$$

Define the transfer function of the system:

$$G(j\omega) = CR(j\omega; A)B \tag{2.190}$$

where $R(j\omega; A)$ is the resolvent operator of A

If we compute the following <u>spectral factorization</u>:

$$(I + G^*G)(\omega) = F^-(j\omega) \cdot F^+(j\omega) \tag{2.191}$$

Then the optimal gain is given by:

$$[B^*k]x_0 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} (F^-(i\omega))^{-1} G^*(i\omega) CR(i\omega; A) x_0 \, d\omega \tag{2.192}$$

In our case

$$A = \begin{bmatrix} 0 & 1 \\ a\frac{\partial^2}{\partial x^2} + b\frac{\partial^2}{\partial y^2} + d & c \end{bmatrix} \tag{2.193}$$

See 1st Chapter for more details.

In the next few sections:

1. We apply this approach to a 1-dimensional scalar problem and compare with the Riccati gain.
2. We compute the optimal gains for the membrane problem assuming a scalar control.
3. We treat the multivariable case.

2.4.1 Comparison Between the Riccati Approach and the Wiener-Hopf Technique for 1-Dimensional Scalar Problem

a) Wiener-Hopf Approach

Consider the system:

$$\begin{cases} \dot{x} = ax + bu \\ x(0) = x_0 \\ y = cx \end{cases} \tag{2.194}$$

Where all quantities belong to $\mathbb{R}$. We assume that $a < 0$ for stability.

Then the gain formula gives:

$$b^* k = \frac{1}{2\pi} \int_{-\infty}^{+\infty} (f^-(j\omega))^{-1} g^*(j\omega) c r(j\omega; a) d\omega \tag{2.195}$$

where $g(j\omega) = cr(j\omega; a)b$

$$\text{hence} \qquad f(j\omega) = 1 + g^* g(j\omega) = 1 + \frac{b^2 c^2}{\omega^2 + a^2} \tag{2.196}$$

This leads to the expression of the spectral factor:

$$f^-(j\omega) = \frac{j\omega - \sqrt{a^2 + b^2 c^2}}{j\omega - a} \tag{2.197}$$

Define: $\alpha = \sqrt{a^2 + b^2 c^2}$

The expression of the gain becomes:

$$bk = \frac{bc^2}{2\pi} \int_{-\infty}^{+\infty} \frac{\omega^2 + a\alpha}{(\omega^2 + \alpha a)^2 + \omega^2(\alpha - a)} d\omega \tag{2.198}$$

b) Riccati Approach:

$$\begin{cases} \dot{x} = ax + bu \\ y = cx \end{cases} \tag{2.199}$$

Performance: $\int_0^\infty \|u\|^2 + \|y\|^2 dt$

The Riccati equation is:

$$k^2 - 2\frac{a}{b^2}k - \frac{c^2}{b^2} = 0 \tag{2.200}$$

the unique positive solution is:

$$k = \frac{a + \sqrt{a^2 + b^2 c^2}}{b^2} \tag{2.201}$$

and therefore:

$$bk = \frac{a + \sqrt{a^2 + b^2 c^2}}{b} \tag{2.202}$$

For the numerical implementation of (2.198) and (2.202) we take:

- $a = -1, b = 1, c = 1$
- precision (absolute and relative) on the computations: 10^{-3}
- Obviously the integral in (2.198) has to be truncated and we need to give an estimate of the spectrum of the transfer function.

The next table gives a comparison between the Riccati equation approach and the Wiener-Hopf approach. The results are excellent if we estimate correctly the spectrum of the transfer function.

Estimate of Spectrum	**10**	**50,000**
Riccati Equation solution	-0.41421	-0.41421
Wiener-Hopf solution	-0.38255	-0.41421

Fig. 10: Comparison Between the Riccati Equation Approach and the Wiener-Hopf Solution for a Scaler Problem

2.4.2 Resolvent Operator Computations:

a) The Resolvent Operator as the Laplace Transform of the Semi-Group:

We quote the following theorem from semi-group theory [72]:

Theorem 2.14:

Let $T(t)$ be a semi-group of class C_0 satisfying: $\|T(t)\| \leq Me^{\omega t}, 0 \leq t \leq \infty$ (as in Theorem 2.2), then the resolvent operator $R(\lambda; A)x = (\lambda I - A)^{-1}x$ is given as the Laplace transform of the semi-group $T(t)$ by:

$$R(\lambda; A) \cdot x = \int_0^\infty e^{-\lambda t} T(t) \cdot x \tag{2.203}$$

where A is the infinitesimal generator of $T(t)$.

The reason why we quote this theorem is to show that one can give a consistent formula for the optimal gains where only the time variable (or the frequency) appears. This way one uses a minimal number of times the Fourier (and inverse Fourier) transform. However we will keep the formula as it is.

b) Associated Elliptic System:

We have $R(s; A) = (sI - A)^{-1}$

$$\text{thus} \quad R(s; A) \begin{bmatrix} \varphi \\ \Psi \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}$$

$$\leftrightarrow (sI - A) \begin{bmatrix} f \\ g \end{bmatrix} = \begin{bmatrix} \varphi \\ \Psi \end{bmatrix} \tag{2.204}$$

$$\text{where} \qquad A \begin{bmatrix} o & 1 \\ a\frac{\partial^2}{\partial x^2} + b\frac{\partial^2}{\partial y^2} + d & c \end{bmatrix} \tag{2.205}$$

We therefore get:

$$\begin{cases} sf - g = \varphi \\ -a\frac{\partial^2 f}{\partial x^2} - b\frac{\partial^2 f}{\partial y^2} - df + (s-c)g = \Psi \end{cases} \tag{2.206}$$

Or in an equivalent form:

$$\begin{cases} a\frac{\partial^2 f}{\partial x^2} + b\frac{\partial^2 f}{\partial y^2} + (d + cs - s^2)f = (c-s)\varphi - \Psi \quad \text{in } \mathring{\Omega} \\ f|_{\partial\Omega} = 0 \\ g = sf - \varphi \end{cases} \tag{2.207}$$

Finally if we take $s = j\omega$ we get:

$$R(jw; A) \begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} \varphi \\ \Psi \end{bmatrix} \tag{2.208}$$

$$\leftrightarrow \begin{cases} a\frac{\partial^2 \varphi}{\partial x^2} + b\frac{\partial^2 \varphi}{\partial y^2} + \{(d + \omega^2) + jc\omega\}\varphi = (c - jw)h - h' \\ \varphi|_{\partial\Omega} = 0 \\ \Psi = j\omega\varphi - h \end{cases} \tag{2.209}$$

If we define $u(x, y, \omega) = (c - j\omega)h - h'$ then the solution of the above PDE is given by:

$$\varphi(x, y, \omega) = \sum_{m=1}^{n} \sum_{n=1}^{n} u(x_m, y_n, \omega) G_{mn}(x, y, \omega) \tag{2.210}$$

2.4.3 First Order Approximation to the Fourier Coefficients of the Control Function

The closed loop control system can be represented by the diagram:

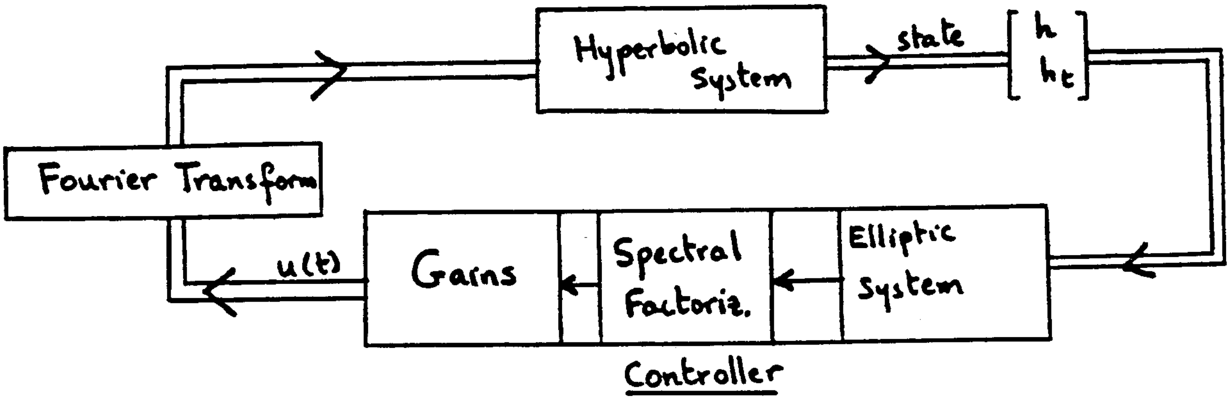


Fig. 11: Controller Block Diagram

The controller computes the control $u(t) = u(x, y, t)$ and we need to compute the Fourier coefficients $u_{mn}(t)$ to feed into the simulator of the membrane equation. Again we are led to the computation of a double integral with oscillatory kernel that can be implemented as a double sum by first order discretization. We know from the theory of double Fourier series that a function $u(x, y)$ and its Fourier coefficients are related by the following formula:

$$\begin{cases} u(x,y) = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \hat{u}_{mn} \sin(\frac{n\pi}{L}x) \sin(\frac{m\pi}{\hat{L}}y) \\ \hat{u}_{mn} = \frac{4}{L\hat{L}} \int_0^L \int_0^{\hat{L}} u(x,y) \sin(\frac{n\pi}{L}x) \sin(\frac{m\pi}{\hat{L}}y) dx dy \end{cases} \tag{2.211}$$

We avoid giving the hypothesis for these formulae to hold and refer the reader to [66]. One needs to do some manipulations to get the above formulae from those in this book. Another word of caution is that only terms of the form $\sin \cdot x \sin \cdot y$ are present in the expansion of $\mu((x, y))$ which is a solution to the PDE and must therefore be given in terms of the eigenfunctions only.

Let us use the usual subdivision of the membrane:

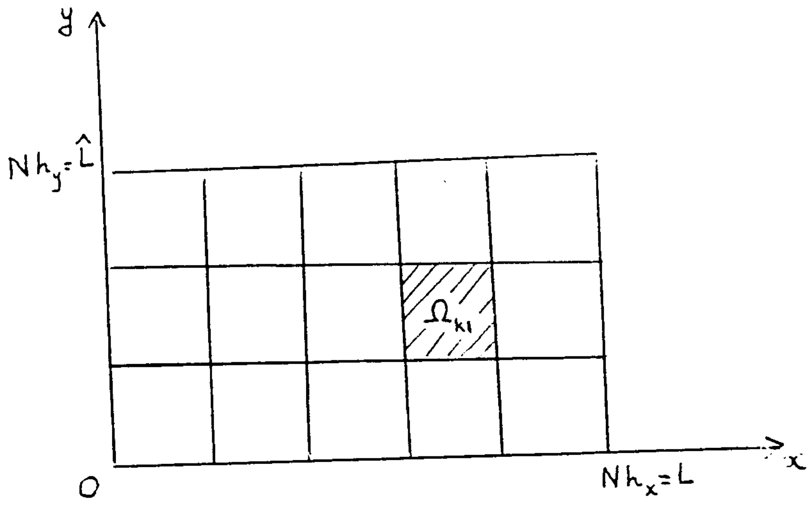


Fig. 12: Grid Points on the Membrane

With $h_x = \frac{L}{N}$ and $h_y = \frac{\hat{L}}{N}$

Where $L, \hat{L}$ are the respective lengths in x, y directions. N the number of subdivisions. h_x, h_y are spatial mesh sizes in x, y directions respectively. The middle point of the rectangle $\Omega_{k\ell} = ((k-1)h_x, kh_x) \times ((\ell-1)h_y, \ell h_y)$ will be referred to as (k, ℓ) and:

$$x_k = (k-1)h_x + \frac{h_x}{2}$$

$$y_\ell = (\ell-1)h_y + \frac{h_y}{2} \tag{2.212}$$

Obviously, we assume that $u(x, y)$ is constant over $\Omega_{k\ell}$ and is given by its value at the point (k, ℓ): $u(x, y) = \mu_{k\ell} \quad (x, y) \in \Omega_{k\ell}$. thus the integral giving $\hat{u}_{mn}$ becomes:

$$\hat{u}_{mn} = \frac{4}{L\hat{L}} \sum_{k=1}^{N} \sum_{\ell=1}^{N} u((k - \frac{1}{2})k_x, (\ell - \frac{1}{2})k_y) \int_{(k-1)h_x}^{kh_x} \sin(\frac{m\pi}{L}x)dx \int_{(\ell-1)h_y}^{\ell h_y}$$

$$\sin(\frac{n\pi}{\hat{L}}y)dy \tag{2.213}$$

which gives:

$$\hat{u}_{mn} = \frac{4}{N^2} \sum_{k=1}^{N} \sum_{\ell=1}^{N} u_{k\ell} \sin(\frac{m\pi}{L}x_k) \sin(\frac{n\pi}{\hat{L}}y_\ell) \tag{2.214}$$

The obvious advantage of this formula is its complete symmetry with one giving $u_{k\ell}$ in terms of $\hat{u}_{mn}$.

In the next page we report the numerical implementation of these formulae and their application to a specific example.

We take $u(x,y) = A\varphi_{mn}(x,y)$ to be the (m,n) eigenfunction

$$\text{i.e.} \qquad \varphi_{mn}(x,y) = \sin(\frac{m\pi}{L}x)\cdot\sin(\frac{n\pi}{\hat{L}}y) \tag{2.215}$$

$N = 8$ is the number of subdivisions

$L = \hat{L} = 1$

$A = 5$ is the amplitude of u

$m = n = 1$ define the desired eignefunction

and we take an expansion with 8 terms (mods). The first matrix is just $(\hat{u}_{mn})$. the second matrix is $F^{-1}(\hat{u}_{mn}) = (u_{k\ell})$. The third matrix is $FF^{-1}(\hat{u}_{mn}) = (\hat{u}_{mn})$ which should coincide with the first matrix. For convenience we have printed only 16 elements of the matrix. The results are satisfying.

A	5.00 0.00 0.00 0.00
	0.00 0.00 0.00 0.00
	0.00 0.00 0.00 0.00
	0.00 0.00 0.00 0.00
$F^{-1}A$	1.94 2.81 2.81 1.94
	2.81 4.06 4.06 2.81
	2.81 4.06 4.06 2.81
	1.94 2.81 2.81 1.94
$FF^{-1}A$	4.92 0.00 0.00 0.00
	0.00 0.00 0.00 0.00
	0.00 0.00 0.00 0.00
	0.00 0.00 0.00 0.00

Fig. 13: Comparison Between the Exact Fourier Coefficients (Matrix A) and the First Order Approximation (Matrix $FF^{-1}A$ Approximating A)

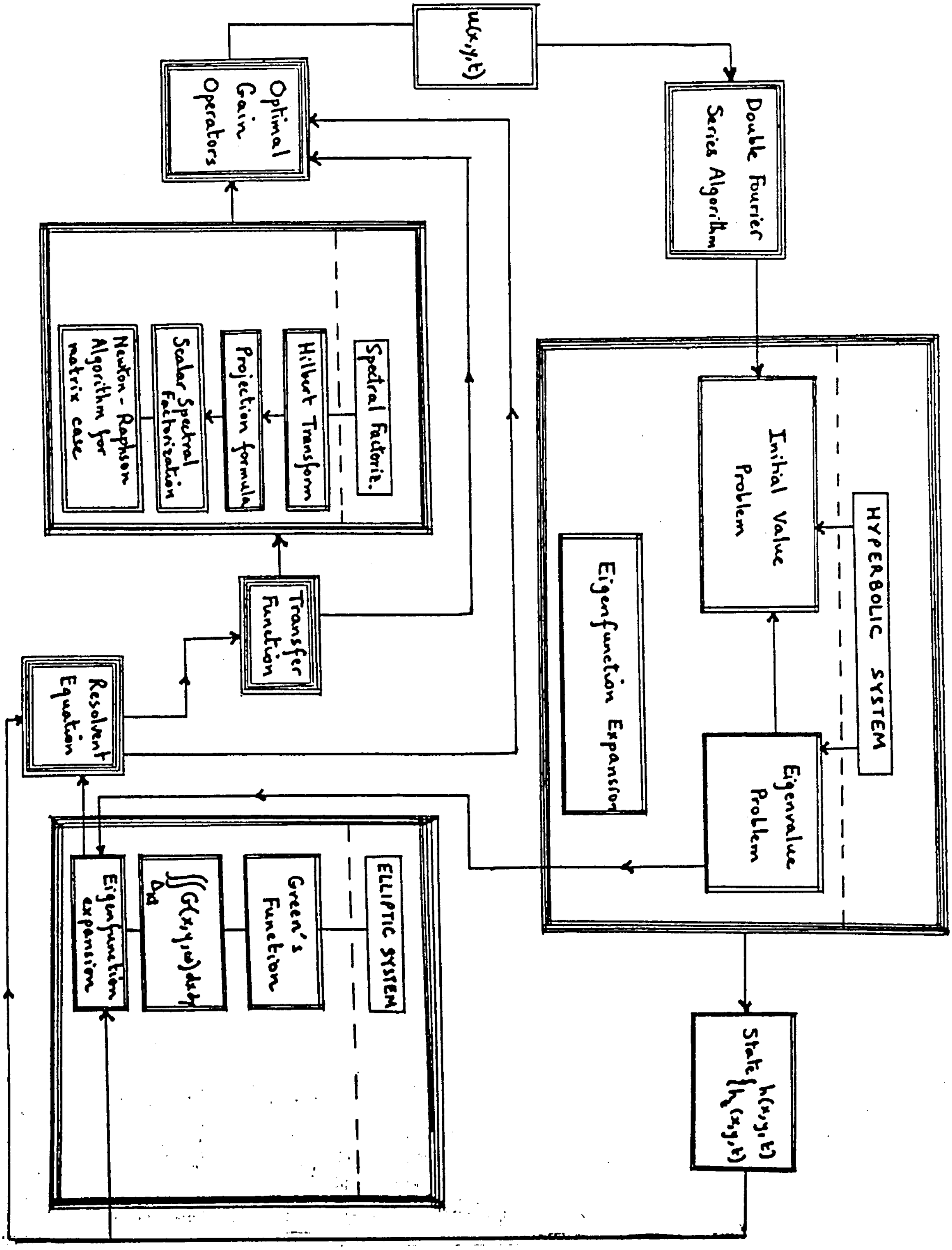


Fig. 14: Block Diagram of the Algorithm

2.4.4 Wiener Hopf Technique for the Scalar Control Problem of the Membrane

We want to compute the optimal control when we take as B operator a scalar multiple of a fixed function (which will be taken to approximate a δ-function). We also observe the membrane at a certain point (x_k, y_ℓ). Thus:

$$\begin{cases} \dot{x} = Ax + u(t)\begin{bmatrix} 0 \\ b_{ij}(x,y) \end{bmatrix} \\ x_0 = x(0) \\ y = Cx = kh(x_k, y_\ell, t) \end{cases} \tag{2.216}$$

where $x = \begin{bmatrix} h \\ h' \end{bmatrix} \in L^2 \oplus L^2$

$$\ell_{ij}(x,y) = \begin{cases} 1 & (x,y) \in \Omega_{ij} \\ 0 & \text{elsewhere} \end{cases} \tag{2.217}$$

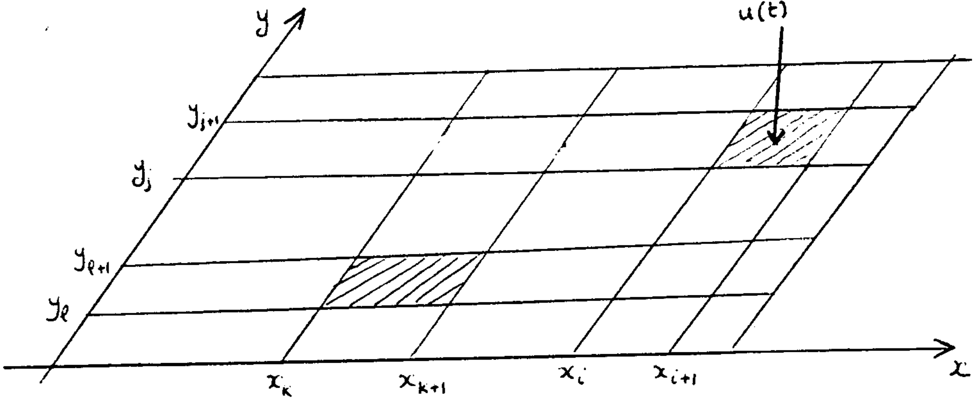


Fig. 15: Membrane With One Controller and One Sensor

a) Transfer Function Computations

$$G(j\omega) = CR(j\omega; A)B \tag{2.218}$$

$$R(j\omega; A)b_{ij}(x,y) = \begin{bmatrix} f_{ij}(x,y) \\ g_{ij}(x,y) \end{bmatrix} \tag{2.219}$$

$$\text{Note that:} \qquad \begin{cases} f_{ij}(x,y,\omega) = +G_{ij}(x,y,\omega) \\ g_{ij}(x,y,\omega) = j\omega f_{ij}(x,y,\omega) \end{cases} \tag{2.220}$$

$$G(j\omega) = C \begin{bmatrix} f_{ij}(x,y) \\ g_{ij}(x,y) \end{bmatrix} = +KG_{ij}(x_k, y_\ell, \omega) \tag{2.221}$$

$$\text{Thus:} \qquad \begin{cases} G(j\omega) = +KG_{ij}(x_k, y_\ell, \omega) \\ G^*(j\omega) = +K\bar{G}_{ij}(x_k, y_\ell, \omega) \end{cases} \tag{2.222}$$

$$F(j\omega) = I + G^*G(j\omega)$$

$$\rightarrow F(j\omega) = 1 + K^2|G_{ij}(x_k, y_\ell, \omega)|^2 \tag{2.223}$$

b) <u>Resolvent Computation</u>:

$$\text{Let} \qquad R(j\omega; A) \begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} \varphi \\ \Psi \end{bmatrix}$$

$$\text{Then} \qquad CR(\omega; A) \begin{bmatrix} h \\ h' \end{bmatrix} = K\varphi(x_k, y_\ell, \omega) \tag{2.224}$$

where

$$\begin{cases} a\frac{\partial^2\varphi}{\partial x^2} + b\frac{\partial^2\varphi}{\partial y^2} + (\omega^2 + jc\omega)\varphi = (c - j\omega)h - h' \\ \varphi|_{\partial\Omega} = 0 \end{cases} \tag{2.225}$$

With the obvious notation: $h_{mn} = h(x_m, y_n), h'_{mn} = h'(x_m, y_n)$

$$\text{Therefore:} \qquad CR(j\omega; A) \begin{bmatrix} h \\ h' \end{bmatrix} = \sum_{m,n=1}^{N} K((c - jw)h_{mn} - h'_{mn})$$

$$G_{mn}(x_k, y_\ell, \omega) \tag{2.226}$$

c) <u>Gain Computations</u>:

$$[B^*K] \begin{bmatrix} h \\ h' \end{bmatrix} = \frac{1}{2\pi} \int_{-\infty}^{+\infty} (F^-(j\omega))^{-1} G^*(j\omega) CR(j\omega; A) \begin{bmatrix} h \\ h' \end{bmatrix} dw$$

$$= \frac{1}{2\pi} \int_{-\infty}^{+\infty} (F^-(j\omega))^{-1} \cdot K \cdot \bar{G}_{ij}(x_k, y_\ell, \omega)$$

$$\sum_{m,n} \underbrace{K((c-j\omega)h_{mn}-h'_{mn})}_{K((ch_{mn}-h'_{mn})-j\omega h_{mn})} G_{mn}(x_k,y_\ell,\omega)d\omega$$

$*$

$$= \frac{K^2}{2\pi}\sum_{m,n}\int_{-\infty}^{+\infty}(F^-(j\omega))^{-1}\bar{G}_{ij}(x_k,y_\ell,\omega)G_{mn}(x_k,y_\ell,\omega)dw$$

$$\cdot (ch_{mn}-h'_{mn})$$

$$-\frac{jK^2}{2\pi}\sum_{m,n}\int_{-\infty}^{+\infty}(F^-(jw))^{-1}\bar{G}_{ij}(x_k,y_\ell,\omega)$$

$$G_{mn}(x_k,y_\ell,\omega)\omega d\omega \cdot h_{mn} \tag{2.227}$$

Define:

$$\begin{cases} I_{mn} = & \frac{1}{2\pi}\int_{-\infty}^{+\infty}(F^-(j\omega))^{-1}\bar{G}_{ij}(x_k,y_\ell,\omega)G_{mn}(x_k,y_\ell,\omega)d\omega \\ \mathcal{J}_{mn} = & -\frac{j}{2\pi}\int_{-\infty}^{+\infty}(F^-(jw))^{-1}\bar{G}_{ij}(x_k,y_\ell,\omega)G_{mn}(x_k,y_\ell,\omega)\omega d\omega \end{cases} \tag{2.228}$$

Therefore we have the following

<u>Theorem 2.15</u>:

The optimal state feedback control for the membrane problem (2.216) is given by:

$$-u = [B^*K]\begin{bmatrix} h \\ h' \end{bmatrix} = K^2\sum_{m,n=1}^{N} I_{mn}(ch_{mn}-h'_{mn}) + \mathcal{J}_{mn}h_{mn}$$

$$= K^2\sum_{m,n=1}^{N}(cI_{mn}+\mathcal{J}_{mn})h_{mn} - I_{mn}h'_{mn} \tag{2.229}$$

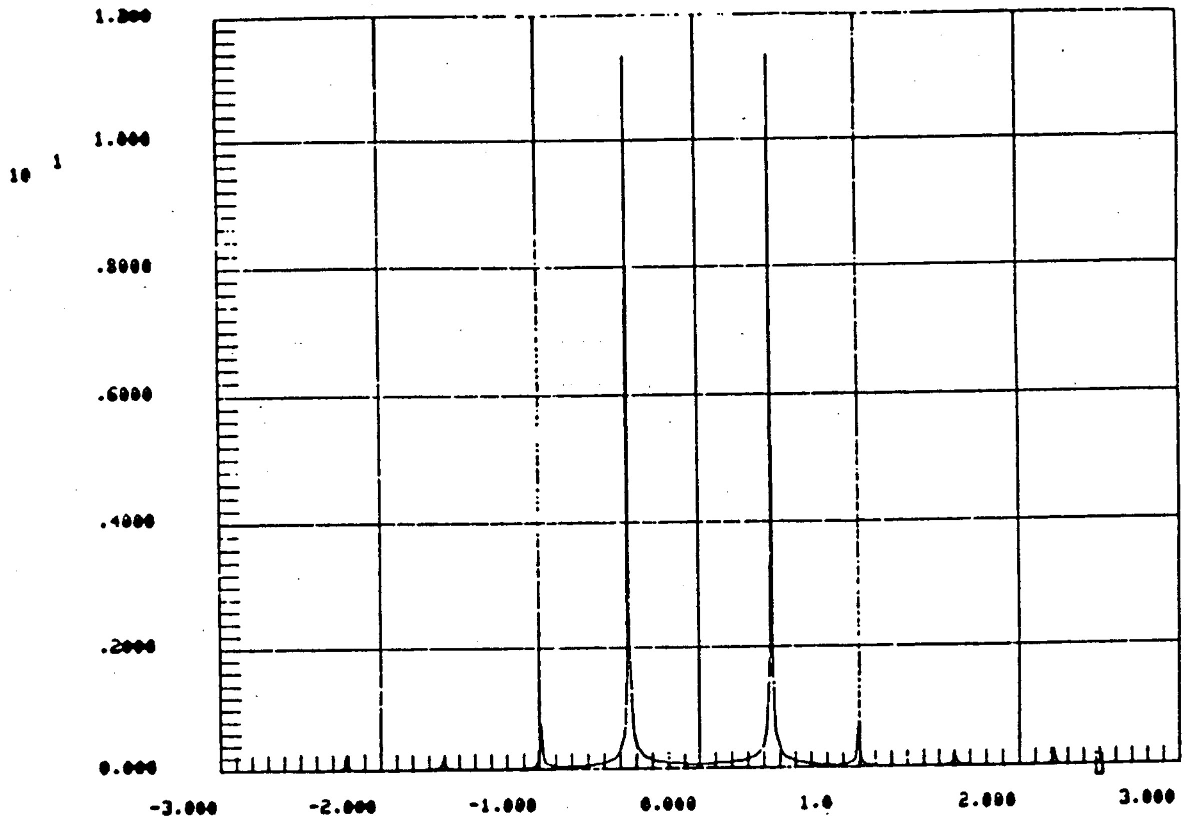
where I_{mn} and J_{mn} are as in (2.228).

The next figures give different graphs of the transfer function for the membrane as follows:

- We choose a point (x,y) on the membrane.

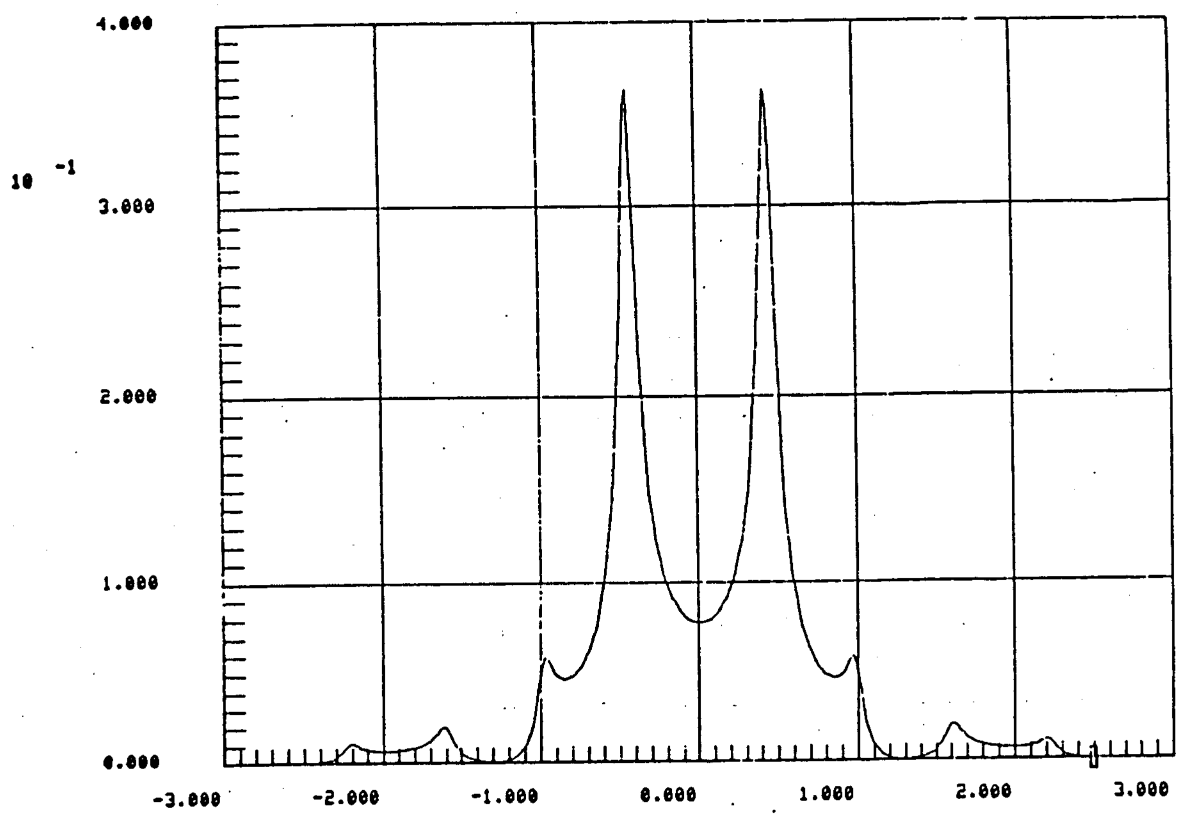
- We assign values to the parameters $a, b, L, \hat{L}$ of the membrane.

The graphs show clearly the influence of the viscous damping c having a smoothing effect on the profile of the transfer function. we can also see the poles at the location of the eigenvalues. These graphs serve only to clarify the above points and make it possible to compare our algorithm to other algorithms by noting the approximate number of dominant poles for instance (see Chapter 1).



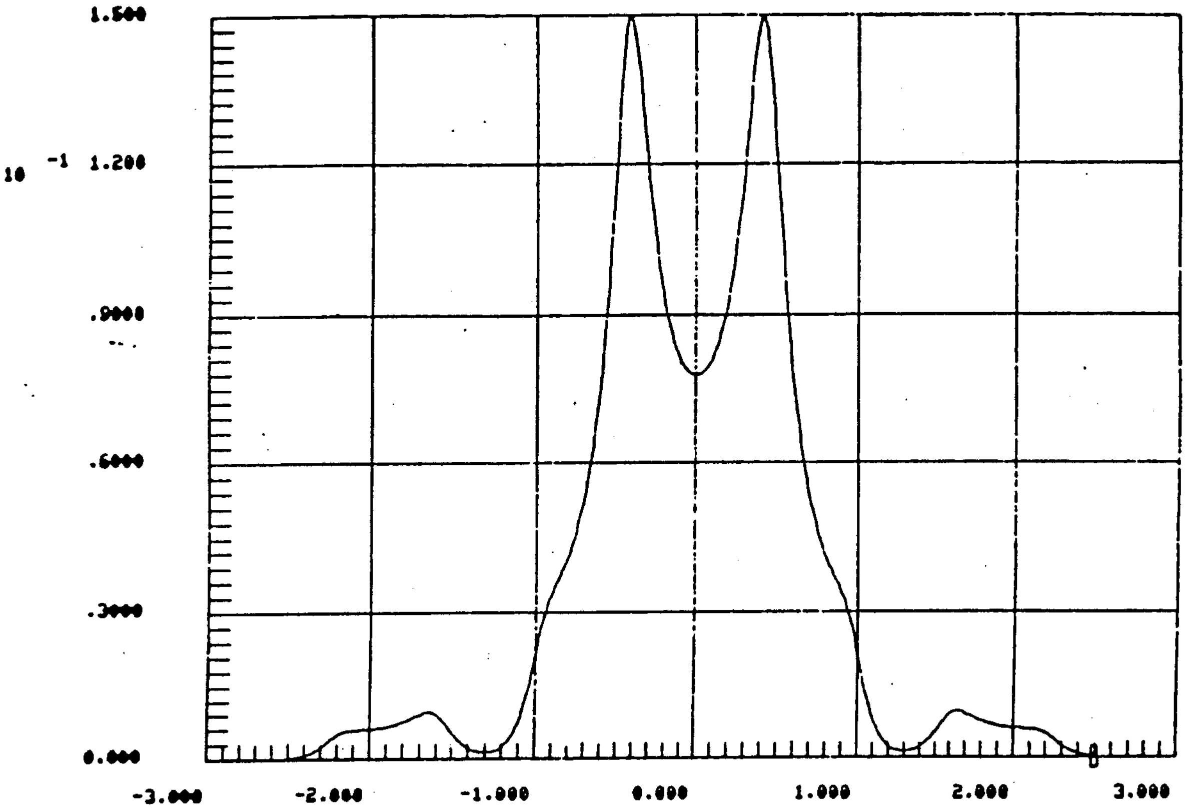
NP = 550 points
a = b = 1
damping: c = 0

Fig. 16a: Transfer Function of a Point on the Membrane With Different Values of the Damping (Which has a Smooting Effect)



NP = 250 points
a = b= 1
damping: c = 1

Fig. 16b: Transfer Function of a Point on the Membrane With Different Values of the Damping (Which has a Smooting Effect)



NP = 350 points
a = b= 1
damping: c = 2.5

Fig. 16c: Transfer Function of a Point on the Membrane With Different Values of the Damping (Which has a Smooting Effect)

The next table shows the performance of the spectral factorization algorithm for the membrane. We take the following values for the parameters (for a description see example of section 2.2.7):

- $L = \hat{L} = 1$
- $m = n = 1$
- $a = b = 1, c = -5, d = 0$
- absolute and relative errors: $10^{-1}, 10^{-1}$
- Number of iterations: 3
- Number of subdivisions: N = 4 (16 grid points)
- Estimated frequency range: $\Omega_0 = 20.0$
- Output gain: K = 10
- Number of points in the frequency domain: NP = 20
- Desired point (x, y) on the membrane where the transfer function is computed: $x = y = 0.3$.

The results are excellent as can be attested from consulting the tables and convergence occurs on the first iteration (see Chapter 1)

Iteration: 1

Frequency: -18

Transfer Function Computed			
1.0004633665	-0.0001383304	0.0004447196	-0.0001383304
0.0000000014	-0.0000653719	0.0000000550	-0.0000653719
-0.0001383288	1.0000543594	-0.0001383290	0.0000539988
0.0000653789	0.00000000001	0.00000653786	0.0000000040
0.0004447196	-0.0001383304	1.0004632473	-0.0001383301
0.0000000550	-0.0000653717	0.0000000014	-0.0000653721
-0.0001383287	0.0000539988	-0.0001383285	1.0000543594
0.0000653788	0.00000000040	0.0000653790	0.0000000001
Exact Transfer Function			
1.0004445314	-0.0001399861	0.0004444980	-0.0001399861
0.	-0.0000655506	0.0000000000	-0.0000655506
-0.0001399861	1.0000537634	-0.0001399861	0.0000537528
0.0000655506	0.	0.0000655506	-0.0000000000
0.0004444980	-0.0001399861	1.0004445314	-0.0001399861
-0.0000000000	-0.0000655506	0.	-0.0000655506
-0.0001399861	0.0000537528	-0.0001399861	1.0000537634
0.0000655506	0.0000000000	0.00006555501	0.0000000000

Fig. 17: Performance of the Spectral Factorization Algorithm: The Exact Transfer Function of the Membrane is Compared with the Transfer Function Based on the Spectral Factor

Iteration: 2

Frequency: -18

Transfer Function Computed			
1.0004446507	-0.0001399860	0.0004445012	-0.0001399859
0.0000000039	-0.0000655508	-0.0000000015	-0.0000655507
-0.0001399858	1.0000538826	-0.001399858	0.0000537527
0.0000655501	0.0000000020	0.0000655500	-0.0000000002
0.0004445012	-0.0001399859	1.0004444122	-0.0001399859
-0.0000000010	-0.0000655507	0.0000000091	-0.0000655507
-0.0001399858	0.0000537527	-0.0001399858	1.0000537634
0.0000655500	0.0000000001	0.0000655500	-0.0000000007
Exact Transfer Function			
1.0004445314	-0.0001399861	0.0004444980	-0.0001399861
0.	-0.0000655506	0.0000000000	-0.0000655506
-0.0001399861	1.0000537634	-0.0001399861	0.0000537528
0.0000655506	0.	0.0000655506	-0.0000000000
0.0004444980	-0.0001399861	1.0004445314	-0.0001399861
-0.0000000000	-0.0000655506	0.	-0.0000655506
-0.0001399861	0.0000537528	-0.0001399861	1.0000537634
0.0000655506	0.0000000000	0.0000655506	0.

Fig. 17 : (Cont'd) Performance of the Spectral Factorization Algorithm: The Exact Transfer Function of the Membrane is Compared with the Transfer Function Based on the Spectral Factor

Iteration: 3

Frequency: -18

Transfer Function Computed			
1.0004446507	-0.0001399861	0.0004444977	-0.0001399861
0.0000000028	-0.0000655510	-0.0000000006	-0.0000655506
-0.0001399862	1.0000537634	-0.001399862	0.0000537528
0.0000655506	0.00000000002	0.0000655506	-0.0000000000
0.0004444978	-0.0001399861	1.0004444122	-0.0001399860
0.0000000004	-0.0000655506	-0.0000000093	-0.0000655505
-0.0001399861	0.0000537528	-0.0001399861	1.0000536442
0.0000655506	0.0000000001	0.0000655506	-0.0000000067
Exact Transfer Function			
1.0004445314	-0.0001399861	0.0004444980	-0.0001399861
0.	-0.0000655506	0.0000000000	-0.0000655506
-0.0001399861	1.0000537634	-0.0001399861	0.0000537528
0.0000655506	0.	0.0000655506	-0.0000000000
0.0004444980	-0.0001399861	1.0004445314	-0.0001399861
-0.0000000000	-0.0000655506	0.	-0.0000655506
-0.0001399861	0.0000537528	-0.0001399861	1.0000537634
0.0000655506	0.0000000000	0.0000655506	0.

Fig. 17: (Cont'd) Performance of the Spectral Factorization Algorithm: The Exact Transfer Function of the Membrane is Compared with the Transfer Function Based on the Spectral Factor

2.4.5 Simulation of the Scalar Optimal State Feedback Control of the Membrane

We refer to the description of Figure 2, section 2.1.4, we give the values of the parameters:

- Absolute and relative errors: 0.01, 0.01
- Time step size: $T = 0.05$
- Parameters of the membrane: $a = 1, b = 1, c = -10$
- Number of grid points and dimensions of the membrane: $N = 5, \ell = 1, \hat{\ell} = 1$
- Index of initial perturbation (eigenfunction) and its amplitude: $m = n = 1, A = 0.1$
- Frequency range estimate: 30.00
- Output gain: $k = 5.0$ The tables compares the displacement and velocity of the membrane under the influence of the control, and the displacement and velocity of the free membrane at different time instants. As we have only one controller (gain $k = 5.0$) on the membrane we see that it takes some time (26 steps at $T = 0.05$ for this example) to bring the membrane to rest, this should be contrasted to the results of Fig. 16 where the action of the control (gains $k = 10$ and $k = 100$) is much faster.

Time = 0

Displacement				Free Displacement			
0.0146	0.0354	0.0354	0.0146	0.0146	0.0354	0.0354	0.0146
0.0354	0.0854	0.0854	0.0354	0.0354	0.0854	0.0854	0.0354
0.0354	0.0854	0.0854	0.0354	0.0354	0.0854	0.0854	0.0354
0.0146	0.0354	0.0354	0.0146	0.0146	0.0354	0.0354	0.0146
Velocity				**Free Velocity**			
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.

Time = 0.1

Displacement				Free Displacement			
0.0136	0.0328	0.0328	0.0136	0.0136	0.0328	0.0328	0.0136
0.0328	0.0792	0.0792	0.0328	0.0328	0.0793	0.0793	0.0328
0.0328	0.0792	0.0792	0.0328	0.0328	0.0793	0.0793	0.0328
0.0136	0.0328	0.0328	0.0136	0.0136	0.0328	0.0328	0.0136
Velocity				**Free Velocity**			
-0.0180	-0.0434	-0.0434	-0.0180	-0.0177	-0.0427	-0.0427	-0.0177
-0.0434	-0.1048	-0.1048	-0.0434	-0.0427	-0.1031	-0.1031	-0.0427
-0.0434	-0.1048	-0.1048	-0.0434	-0.0427	-0.1031	-0.1031	-0.0427
-0.0180	-0.0434	-0.0434	-0.0180	-0.0177	-0.0427	-0.0427	-0.0177

Fig. 18: Results of the Simulation on the Scalar Optimal State Feedback Control of the Membrane

Time = 35

Displacement				Free Displacement			
0.0082	0.0198	0.0198	0.0082	0.0084	0.0202	0.0202	0.0084
0.0198	0.0478	0.0478	0.0198	0.0202	0.0487	0.0487	0.0202
0.0198	0.0478	0.0478	0.0198	0.0202	0.0487	0.0487	0.0202
0.0082	0.0198	0.0198	0.0082	0.0084	0.0202	0.0202	0.0084
Velocity				**Free Velocity**			
-0.0202	-0.0489	-0.0489	-0.0202	-0.0195	-0.0472	-0.0472	-0.0195
-0.0489	-0.1179	-0.1179	-0.0489	-0.0472	-0.1138	-0.1138	-0.0472
-0.0489	-0.1179	-0.1179	-0.0489	-0.0472	-0.1138	-0.1138	-0.0472
-0.0202	-0.0489	-0.0489	-0.0202	-0.0195	-0.0472	-0.0472	-0.0195

Time = 0.45

Displacement				Free Displacement			
0.0063	0.0153	0.0153	0.0063	0.0066	0.0158	0.0158	0.0066
0.0153	0.0369	0.0369	0.0153	0.0158	0.0383	0.0383	0.0158
0.0153	0.0369	0.0369	0.0153	0.0158	0.0383	0.0383	0.0158
0.0063	0.0153	0.0153	0.0063	0.0066	0.0158	0.0158	0.0066
Velocity				**Free Velocity**			
-0.0171	-0.0413	-0.0413	-0.0171	-0.0163	-0.0393	-0.0393	-0.0163
-0.0413	-0.0997	-0.0997	-0.0413	-0.0393	-0.0949	-0.0949	-0.0393
-0.0413	-0.0997	-0.0997	-0.0413	-0.0393	-0.0949	-0.0949	-0.0393
-0.0171	-0.0413	-0.0413	-0.0171	-0.0163	-0.0393	-0.0393	-0.0163

Fig. 18: (Cont'd) Results of the Simulation on the Scalar Optimal State Feedback Control of the Membrane

Time = 0.6

Displacement				Free Displacement			
0.0041	0.0099	0.0099	0.0041	0.0045	0.0108	0.0108	0.0045
0.0099	0.0239	0.0239	0.0099	0.0108	0.0261	0.0261	0.0108
0.0099	0.0239	0.0239	0.0099	0.0108	0.0261	0.0261	0.0108
0.0041	0.0099	0.0099	0.0041	0.0045	0.0108	0.0108	0.0045
Velocity				**Free Velocity**			
-0.0126	-0.0305	-0.0305	-0.0126	-0.0116	-0.0281	-0.0281	-0.0116
-0.0305	-0.0736	-0.0736	-0.0305	-0.0281	-0.0678	-0.0678	-0.0281
-0.0305	-0.0736	-0.0736	-0.0305	-0.0281	-0.0678	-0.0678	-0.0281
-0.0126	-0.0305	-0.0305	-0.0126	-0.0116	-0.0281	-0.0281	-0.0116

Time = 0.85

Displacement				Free Displacement			
0.0016	0.0039	0.0039	0.0016	0.0023	0.0056	0.0056	0.0023
0.0039	0.0095	0.0095	0.0039	0.0056	0.0135	0.0135	0.0056
0.0039	0.0095	0.0095	0.0039	0.0056	0.0135	0.0135	0.0056
0.0016	0.0039	0.0039	0.0016	0.0023	0.0056	0.0056	0.0023
Velocity				**Free Velocity**			
-0.0073	-0.0177	-0.0177	-0.0073	-0.0062	-0.0149	-0.0149	-0.0062
-0.0177	-0.0426	-0.0426	-0.0177	-0.0149	-0.0361	-0.0361	-0.0149
-0.0177	-0.0426	-0.0426	-0.0177	-0.0149	-0.0361	-0.0361	-0.0149
-0.0073	-0.0177	-0.0177	-0.0073	-0.0062	-0.0149	-0.0149	-0.0062

Fig. 18: (Cont'd) Results of the Simulation on the Scalar Optimal State Feedback Control of the Membrane

Time = 95

Displacement				Free Displacement			
0.0010	0.0023	0.0023	0.0010	0.0018	0.0043	0.0043	0.0018
0.0023	0.0056	0.0056	0.0023	0.0043	0.0103	0.0103	0.0043
0.0023	0.0056	0.0056	0.0023	0.0043	0.0103	0.0103	0.0043
0.0010	0.0023	0.0023	0.0010	0.0018	0.0043	0.0043	0.0018
Velocity				**Free Velocity**			
-0.0056	-0.0136	-0.0136	-0.0056	-0.0048	-0.0115	-0.0115	-0.0048
-0.0136	-0.0328	-0.0328	-0.0136	-0.0115	-0.0277	-0.0277	-0.0115
-0.0136	-0.0328	-0.0328	-0.0136	-0.0115	-0.0277	-0.0277	-0.0115
-0.0056	-0.0136	-0.0136	-0.0056	-0.0048	-0.0115	-0.0115	-0.0048

Time = 1.25

Displacement				Free Displacement			
0.0000	0.0001	0.0001	0.0000	0.0008	0.0019	0.0019	0.0008
0.0019	0.0046	0.0046	0.0019	0.0019	0.0047	0.0047	0.0019
0.0019	0.0046	0.0046	0.0019	0.0019	0.0047	0.0047	0.0019
0.0000	0.0001	0.0001	0.0000	0.0008	0.0019	0.0019	0.0008
Velocity				**Free Velocity**			
-0.0008	-0.0019	-0.0019	-0.0008	-0.0021	-0.0051	-0.0051	-0.0021
-0.0019	-0.0047	-0.0047	-0.0019	-0.0051	-0.0124	-0.0124	-0.0051
-0.0019	-0.0047	-0.0047	-0.0019	-0.0051	-0.0124	-0.0124	-0.0051
-0.0008	-0.0019	-0.0019	-0.0008	-0.0021	-0.0051	-0.0051	-0.0021

Fig. 18: (Cont'd) Results of the Simulation on the Scalar Optimal State Feedback Control of the Membrane

Time = 1.3

Displacement				Free Displacement			
0.0000	0.0000	0.0000	0.0000	0.0007	0.0017	0.0017	0.0007
0.0000	0.0000	0.0000	0.0000	0.0017	0.0040	0.0040	0.0017
0.0000	0.0000	0.0000	0.0000	0.0017	0.0040	0.0040	0.0017
0.0000	0.0000	0.0000	0.0000	0.0007	0.0017	0.0017	0.0007
Velocity				**Free Velocity**			
-0.0003	-0.0007	-0.0007	-0.0003	-0.0019	-0.0045	-0.0045	-0.0019
-0.0007	-0.0018	-0.0018	-0.0007	-0.0045	-0.0109	-0.0109	-0.0045
-0.0007	-0.0018	-0.0018	-0.0007	-0.0045	-0.0109	-0.0109	-0.0045
-0.0003	-0.0007	-0.0007	-0.0003	-0.0019	-0.0045	-0.0045	-0.0019

Fig. 18: (Cont'd) Results of the Simulation on the Scalar Optimal State Feedback Control of the Membrane

2.4.6 Multidimensional Control-Multidimensional Observation of the Vibrating Membrane

In this paragraph we assume that the membrane can be controlled at discrete points evenly distributed for convenience, and that we observe the displacement of the whole membrane. From a computational point of view it is enough to assume that the membrane is obseved at a discrete number of points.

Thus we have the following system:

$$\begin{cases} \dot{X} = AX + \sum_{i,j=1}^{N} u_{ij}(t) \begin{bmatrix} 0 \\ b_{ij}(x,y) \end{bmatrix} \\ y = CX = C \begin{bmatrix} h \\ h' \end{bmatrix} = K \begin{bmatrix} h(x_1, y_1) \\ \vdots \\ h(x_N, y_n) \end{bmatrix} \end{cases} \tag{2.230}$$

We note that $b_{ij}(x,y) = \begin{cases} 1 & (x,y) \in \Omega_{ij} \\ 0 & \text{otherwise} \end{cases}$

with $x_1, x_2, \dots x_N$ a subdivision of $[0, L]$ on $x-$axis,

$y_1, y_2, \dots, y_N$ a subdivision of $[0, \hat{L}]$ on $y-$axis.

K is a scalar gain that gives the weight of the observation relative to the control in the performance criterion:

$$\int_0^\infty \|u(t)\|^2 + K^2 \|h(t)\|^2 dt \tag{2.231}$$

a) Transfer Function:

$$G(j\omega) = CR(j\omega; A)B$$

$$\langle G(j\omega)[u_{ij}], [v_{ij}] \rangle_{\mathbb{C}^{N^2}} = \langle [u_{ij}], G^*(j\omega)[v_{ij}] \rangle_{\mathbb{C}^{N^2}}$$

$$G(j\omega)[u_{ij}] = CR(j\omega; A)B[u_{ij}] = CR(j\omega; A) \sum_{i,j=1}^{N} u_{ij} \begin{bmatrix} 0 \\ b_{ij}(x,y) \end{bmatrix}$$

$$= C \sum_{i,j=1}^{N} u_{ij} R(j\omega; A) \begin{bmatrix} 0 \\ b_{ij}(x,y) \end{bmatrix}$$

$$\Rightarrow G(j\omega)\,[u_{ij}] = K \begin{bmatrix} G_{ij}(x_1, y_1, \omega) \\ \vdots \\ G_{ij}(x_N, y_n, \omega) \end{bmatrix} \tag{2.232}$$

where $G_{ij}(x_k, y_\ell, \omega)$ is the integrated discrete Green's function on Ω_{ij} evaluated at (x_k, y_ℓ).

The final formula for G is:

$$G(j\omega)[u_{ij}] = K \sum_{i,j=1}^{N} u_{ij} \begin{bmatrix} G_{ij}(x_1, y_1, \omega) \\ \vdots \\ G_{ij}(x_N, Y_N, \omega) \end{bmatrix} \in \mathbb{C}^{N^2} \tag{2.233}$$

Using the duality relation between G and G^* one can prove in a similar fashion that if:

$$G^*(j\omega)[v_{k\ell}] = [w_{ij}] \tag{2.234}$$

and w_{ij} are given by

$$w_{ij} = K \sum_{k=1}^{N} \sum_{\ell=1}^{N} \bar{G}_{ij}(x_k, y_\ell, \omega) v_{k\ell} \tag{2.235}$$

where the $\bar{G}_{ij}(x_k, y_\ell, \omega)$ is the complex conjugate of $G_{ij}(x_k, y_\ell, \omega)$. Hence it is possible to compute:

$$F(j\omega) = I + G^* G(j\omega) \tag{2.236}$$

Let

$$F(j\omega)[u_{ij}] = [w_{ij}] = I + G^* G[u_{ij}] \tag{2.237}$$

We have that:

$$G^* G[u_{ij}] = G^* \left\{ K \sum_{i,j=1}^{N} u_{ij} [G_{ij}(x_k, y_\ell, \omega)]_{k\ell} \right\} \tag{2.238}$$

where $[G_{ij}(x_k, y_\ell, \omega)]_{k\ell}$ means the vector $\begin{bmatrix} G_{ij}(x_1, y_1, \omega) \\ \vdots \\ G_{ij}(x_N, y_N, \omega) \end{bmatrix}$ and we are using the canonical isomorphism between $\mathbb{C}^{N^2}$ and $\mathcal{L}(\mathbb{C}^N, \mathbb{C}^N)$ the vector space of linear maps from $\mathbb{C}^N$ to $\mathbb{C}^N$. $\mathbb{C}^{N^2}$ is the linear space of vectors of N^2 complex entries.

Therefore:

$$\begin{aligned} G^*G[u_{ij}] &= K \sum_{i,j=1}^{N} u_{ij} G^*[G_{ij}(x_k, y_\ell, \omega)]_{k\ell} \\ &= K \sum_{i,j=1}^{N} u_{ij} \left\{ K \sum_{k,\ell=1}^{N} G_{ij}(x_k, y_\ell, \omega)[\bar{G}_{mn}(x_k, y_\ell, \omega)]_{mn} \right\} \qquad (2.239) \\ &\quad \text{(by (37))} \end{aligned}$$

Or in a more convenient notation:

$$G^*G(j\omega)[u_{ij}] = K^2 \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{N} \sum_{\ell=1}^{N} u_{ij} G_{ij}(x_k, y_\ell, \omega) \begin{bmatrix} \bar{G}_{11}(x_k, y_\ell, \omega) \\ \vdots \\ \bar{G}_{mn}(x_k, y_\ell, \omega) \\ \vdots \\ \bar{G}_{NN}(x_k, y_\ell, \omega) \end{bmatrix} \qquad (2.240)$$

If we set:

$$[w_{k\ell}] = G^*G[u_{ij}] \qquad (2.241)$$

Then

$$w_{k\ell} = K^2 \sum_{i,j=1}^{N} \sum_{m,n=1}^{N} u_{ij} G_{ij}(x_m, y_n, \omega) \bar{G}_{k\ell}(x_m, y_n, \omega) \qquad (2.242)$$

It is obvious at this stage that a successful implementation of these formulae will depend in an essential way on how to handle this multitude of indices, and how to represent the vector $[G_{ij}(x_k, y_\ell, \omega)]_{k\ell}$ i.e. how to order the elements of this $N \times N$ vector (which in fact represents an operator from $\mathbb{C}^N \to \mathbb{C}$.)

The problem of ordering the indices is the same problem as counting the rational numbers, which could be achieved using the diagonalization algorithm:

$$i = 1, 2, \ldots N \qquad j = 1, 2, \ldots N$$

$$\alpha = (i, j) = 1, 2, 3, \ldots N^2 \tag{2.243}$$

From now on we use greek letters to designate pairs of indices. We do not use the diagonalization algorithm which is of theoretical value only, the following simpler counting process turned out to be more useful (in easiness of implementation). It consists by starting in the first row (and exhausting all elements of the matrix) then the 2nd row, etc.

To get i and j from $\alpha = (i, j)$ you apply the Euclidean division algorithm (modified):

$$\alpha = (i, j) \longrightarrow \alpha - 1 = (i-1)N + (j-1) 0 \leq i - 1 \leq N - 1$$
$$0 \leq j - 1 \leq N - 1 \tag{2.244}$$

Hence

$$\begin{cases} j - 1 = \text{mod }(\alpha - 1, N) \\ i - 1 = (\alpha - j)/N \end{cases} \tag{2.245}$$

and

$$\text{Conversely } \alpha = (i - 1)N + j \tag{2.246}$$

With this notation in mind let us define:

$$\begin{aligned} \alpha &= (i, j) \\ \beta &= (k, \ell) \\ \gamma &= (m, n) \\ \text{and} \qquad P_\gamma &= (x_m, y_n) \end{aligned} \tag{2.247}$$

Then the expression (108) becomes:

$$w_\beta = \sum_{\alpha=1}^{N^2} (K^2 \sum_{\gamma=1}^{N^2} \bar{G}_\beta(P_\gamma, \omega) \cdot G_\alpha(P_\gamma, \omega)) u_\alpha \tag{2.248}$$

$$\text{or} \qquad w_\beta = \sum_{\alpha=1}^{N^2} A_{\beta\alpha} u_\alpha \tag{2.249}$$

$$\text{where} \qquad A_{\beta\alpha} = K^2 \sum_{\gamma=1}^{N^2} \bar{G}_\beta(P_\gamma,\omega) \cdot G_\alpha(P_\gamma,\omega) \tag{2.250}$$

In other words $A_{\alpha\beta}$ is the matrix representation of the operator G^*G:

$$[G^*G(\omega)] = [A_{\beta\alpha}] \tag{2.251}$$

$$\text{Now} \qquad F(j\omega) = I + G^*G(j\omega) = [F_{\beta\alpha}] \tag{2.252}$$

therefore

$$F_{\beta\alpha} = \delta_\beta^\alpha + K^2 \sum_{\gamma=1}^{N^2} \bar{G}_\beta(P_\gamma,\omega) \cdot G_\alpha(P_\gamma,\omega) \tag{2.253}$$

Expression of the resolvent operator:

To see where the following expression comes from refer to section 2:

$$CR(j\omega; A) \begin{bmatrix} h \\ h' \end{bmatrix} = \sum_{p=1}^{N} \sum_{q=1}^{N} K \left\{ (C - j\omega) h_{pq} - h'_{pq} \right\} \begin{bmatrix} G_{pq}(x_1, y_1, \omega) \\ \vdots \\ G_{pq}(x_N, y_N, \omega) \end{bmatrix} \tag{2.254}$$

Gain Formulae:

$$B^*K \begin{bmatrix} h \\ h' \end{bmatrix} = \frac{1}{2\pi} \int_{-\infty}^{+\infty} [F^-(j\omega)]^{-1} \cdot G^*(j\omega) CR(j\omega; A) \begin{bmatrix} h \\ h' \end{bmatrix} d\omega \tag{2.255}$$

$$G^*(j\omega) \cdot CR(j\omega; A) \begin{bmatrix} h \\ h' \end{bmatrix} = G^*(j\omega) \left\{ \sum_{p=1}^{N} \sum_{q=1}^{N} K((c - j\omega) h_{pq} - h'_{pq}) \right\} \begin{bmatrix} G_{pq}(x_1, y_1, \omega) \\ \vdots \\ G_{pq}(x_N, y_N, \omega) \end{bmatrix} \tag{2.256}$$

Using the linearity of G^*, (118) becomes:

$$G^*(j\omega)\cdot\begin{bmatrix} \sum_{p=1}^N \sum_{q=1}^N K((c-j\omega)h_{pq}-h'_{pq})G_{pq}(x_1,y_1,\omega) \\ \vdots \\ \sum_{p=1}^N \sum_{q=1}^N K((c-j\omega)h_{pq}-h'_{pq})G_{pq}(x_k,y_\ell,\omega) \\ \vdots \\ \sum_{p=1}^N \sum_{q=1}^N K((c-j\omega)h_{pq}-h'_{pq})G_{pq}(x_N,y_N,\omega) \end{bmatrix} \tag{2.257}$$

By using (103) expression (120) becomes:

$$= \begin{bmatrix} K\sum_{k,\ell=1}^N \bar{G}_{11}(x_k,y_\ell,\omega)\cdot\sum_{p=1}^N\sum_{q=1}^N K((c-jw)h_{pq}-h'_{pq}) \\ G_{pq}(x_k,y_\ell,\omega) \\ \vdots \\ K\sum_{k,\ell=1}^N \bar{G}_{ij}(x_k,y_\ell,\omega)\cdot\sum_{p=1}^N\sum_{q=1}^N K((c-j\omega)h_{pq}-h'_{pq}) \\ \cdot G_{pq}(x_k,y_\ell,\omega) \\ \vdots \\ K\sum_{k=1\ell=1}^N\sum_{\ell=1}^N \bar{G}_{NN}(x_k,y_\ell,\omega)\sum_{p=1}^N\sum_{q=1}^N K((c-j\omega)h_{pq}-h'_{pq}) \\ \cdot G_{pq}(x_k,y_\ell,\omega) \end{bmatrix} \tag{2.258}$$

Let us define the new quantities:

$$R_{k\ell}(\omega)=\sum_{p=1}^{N}\sum_{q=1}^{N}\left\{(c-j\omega)h_{pq}-h'_{pq}\right\}G_{pq}(x_k,y_\ell,\omega) \tag{2.259}$$

to get:

$$G^*(j\omega)CR(j\omega;A)\begin{bmatrix} h \\ h' \end{bmatrix} = K^2\sum_{k=1}^{N}\sum_{\ell=1}^{N}R_{k\ell}(\omega)[\bar{G}_{ij}(x_k,y_\ell,\omega)]_{ij} \tag{2.260}$$

Hence:

$$B^*K\begin{bmatrix} h \\ h' \end{bmatrix} = \frac{K^2}{2\pi}\int_{-\infty}^{+\infty}[F^-(j\omega)]^{-1}\sum_{k=1}^{N}\sum_{\ell=1}^{N}R_{k\ell}(\omega)[\bar{G}_{ij}(x_k,y_\ell,\omega)]_{ij}d\omega \tag{2.261}$$

By linearity we have:

$$B^*K\begin{bmatrix} h \\ h' \end{bmatrix} = \frac{K^2}{2\pi}\sum_{k=1}^{N}\sum_{p=1}^{N}\sum_{-\infty}^{+\infty}R_{k\ell}(\omega)\cdot\left\{F^-(j\omega)^{-1}\cdot[\bar{G}_{ij}(x_k,y_\ell,\omega)]_{ij}\right\}d\omega \tag{2.262}$$

Define the vector $\vec{V}\ k\ell(\omega)$ by:

$$\vec{V}_{k\ell}\,(\omega) = [F^{-}(j\omega)]^{-1} \cdot \begin{bmatrix} \bar{G}_{11}(x_k, y_\ell, \omega) \\ \vdots \\ \bar{G}_{ij}(x_k, y_\ell, \omega) \\ \vdots \\ \bar{G}_{NN}(x_k, y_\ell, \omega) \end{bmatrix} \tag{2.263}$$

To get the following

Theorem 2.16:

The optimal state feedback control for the multivariable problem of the membrane is given by:

$$[B^{*}K]\begin{bmatrix} h \\ h' \end{bmatrix} = \frac{K^2}{2\pi}\int_{-\infty}^{+\infty}\sum_{k=1}^{N}\sum_{\ell=1}^{N} R_{k\ell}(\omega)\cdot \vec{V}_{k\ell}\,(\omega)d\omega \tag{2.264}$$

where $R_{k\ell}(\omega)$ and $V_{k\ell}(\omega)$ are given by (2.2.59) and (2.263).

2.4.7 Control of the Membrane Along a Mode of the System

In this section we investigate the control problem when the input is a time varying function multiple of a mode of the system, while we observe the displacement of the whole membrane. Thus:

$$\begin{cases} \dot{x} = Ax + \{\, u_{ij}(t)\ \varphi_{ij}(x,y)\,\} \\ y = Cx \end{cases}$$

$$B : \mathbb{C} \to L^2 \oplus L^2$$

$$u_{ij}(t) \to u_{ij}(t)\begin{bmatrix} 0 \\ \varphi_{ij}(x,y) \end{bmatrix}$$

$$C : L^2 \oplus L^2 \to L^2$$

$$\begin{bmatrix} f \\ g \end{bmatrix} \to Kf \tag{2.265}$$

where $\varphi_{ij}(x,y)$ is the eigenfunction corresponding to the eigenvalue

$$\lambda_{ij} = -\left\{ a\left(\frac{i\pi}{L}\right)^2 + b\left(\frac{j\pi}{\hat{L}}\right)^2 \right\} \tag{2.266}$$

K is the weight of the output relative to the control in the performance criterion:

$$J = \int_0^\infty \|u(t)\|^2 + K^2\|h(t)\|^2 dt \tag{2.267}$$

Resolvent Computations

We know that:

$$R(j\omega; A)\begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix} \qquad \text{if and only if:} \tag{2.268}$$

$$\begin{cases} a\frac{\partial^2 f}{\partial x^2} + b\frac{\partial^2 f}{\partial y^2} + (d + \omega^2 + jc\omega)f = (c - j\omega)h - h' \\ f|_{\partial\Omega} = 0 \\ g = j\omega f - h \end{cases} \tag{2.269}$$

Let us assume that:

$$\begin{aligned} & h(x,y,t) = \alpha(t)\varphi_{ij}(x,y) \\ \text{and} \quad & h'(x,y,t) = \beta(t)\varphi_{ij}(x,y) \end{aligned} \tag{2.270}$$

where $\varphi_{ij}(x,y)$ is the (i,j) eigenfunction. Then the solution to the above PDE is:

$$f(x,y,t) = \frac{\alpha(t)(c - j\omega) - \beta(t)}{(\lambda_{ij} + (d + \omega^2)) + jc\omega}\varphi_{ij}(x,y) \tag{2.271}$$

Transfer Function:

$$G(j\omega) = CR(j\omega; A)B \tag{2.272}$$

$$\begin{aligned} G(j\omega)u_{ij}(t) &= CR(j\omega; A)\begin{bmatrix} 0 \\ u_{ij}\varphi_{ij}(x,y) \end{bmatrix} = C\begin{bmatrix} f_{ij}(x,y,\omega) \\ g_{ij}(x,y,\omega) \end{bmatrix} \\ &= Kf_{ij}(x,y,\omega) \end{aligned} \tag{2.273}$$

The function $f_{ij}(z, y, \omega)$ is the solution to the PDE:

$$\begin{cases} a\frac{\partial^2 f_{ij}}{\partial x^2} + b\frac{\partial^2 f_{ij}}{\partial y^2} + (d + \omega^2 + jc\omega)f_{ij} = -u_{ij}(t)\varphi_{ij}(x, y) \\ f_{ij}|_{\partial\Omega} = 0 \end{cases} \tag{2.274}$$

Let us look for a solution of the form

$$f_{ij}(x, y, \omega) = \chi(\omega)\varphi_{ij}(x, y) \tag{2.275}$$

Then plugging into the PDE and using the fact that $\varphi_{ij}(x, y)$ is an eigenfunction gives:

$$G(j\omega)u_{ij}(t) = \frac{K\varphi_{ij}(x, y)}{(\lambda_{ij} + d + \omega^2) + jc\omega}u_{ij}(t) \tag{2.276}$$

Similarly if $\Psi \in L^2$, then:

$$G^*(j\omega)\Psi = -\frac{K}{(\lambda_{ij} + d + \omega^2) - jc\omega}\iint_{\Omega} \varphi_{ij}(\zeta, \eta)\Psi(\zeta, \eta)d\zeta dy \tag{2.277}$$

Therefore the expression of the modified transfer function is:

$$F(j\omega) = I + G^*G(j\omega) = 1 + \frac{K^2}{(\lambda_{ij} + d + \omega^2)^2 + c^2\omega^2} \cdot \int\int_{\Omega} |\varphi(\zeta, \eta)|^2 d\zeta dy \tag{2.278}$$

<u>Gain computation</u>:

The expression of the gain is:

$$B^*K\begin{bmatrix} h \\ h' \end{bmatrix} = \frac{1}{2\pi}\int_{-\infty}^{+\infty} (F^-(j\omega))^{-1}G^*(j\omega)CR(j\omega; A)\begin{bmatrix} h \\ h' \end{bmatrix} d\omega \tag{2.279}$$

Plugging (133), (138) into (140) gives:

$$B^*K\begin{bmatrix} h \\ h' \end{bmatrix} = -\frac{K^2}{2\pi}\iint_{\Omega} |\varphi(\zeta, \eta)^2 d\zeta d\eta \cdot \int_{-\infty}^{+\infty} [F^-(j\omega)]^{-1}$$

$$\cdot \frac{\alpha(t)(c - jw) - \beta(t)}{(\lambda_{ij} + d + \omega^2)^2 + c^2\omega^2}d\omega \tag{2.280}$$

Theorem 2.17:

Define:

$$I = \int_{-\infty}^{+\infty} F^{-}(j\omega)^{-1} \frac{c - jw}{(\lambda_{ij} + d + \omega^2)^2 + c^2\omega^2} d\omega$$

$$J = \int_{-\infty}^{+\infty} F^{-}(j\omega)^{-1} \frac{d\omega}{(\lambda_{ij} + d + \omega^2)^2 + c^2\omega^2} \tag{2.281}$$

Then the optimal state feedback control for the membrane problem (2.265) is given by:

$$u(t) = \frac{K^2}{2\pi} \iint_{\Omega} |\varphi(\zeta, \eta)|^2 d\zeta d\eta \cdot (I\alpha(t) - J\beta(t)) \tag{2.282}$$

where $\varphi(\zeta, n)$ is the desired mode shape.

Next we report the simulation of this type of controller. We choose a point (x, y) on the membrane and we consider its motion in the following conditions:

- the membrane is subject to free vibrations.
- the membrane is under influence of the control.

starting from the same initial conditions we see clearly that the membrane goes much faster to rest when under the influence of the controller. We also see that as the gain gest higher (from $k = 10$ to $k = 100$) the membrane is brought to rest much faster.

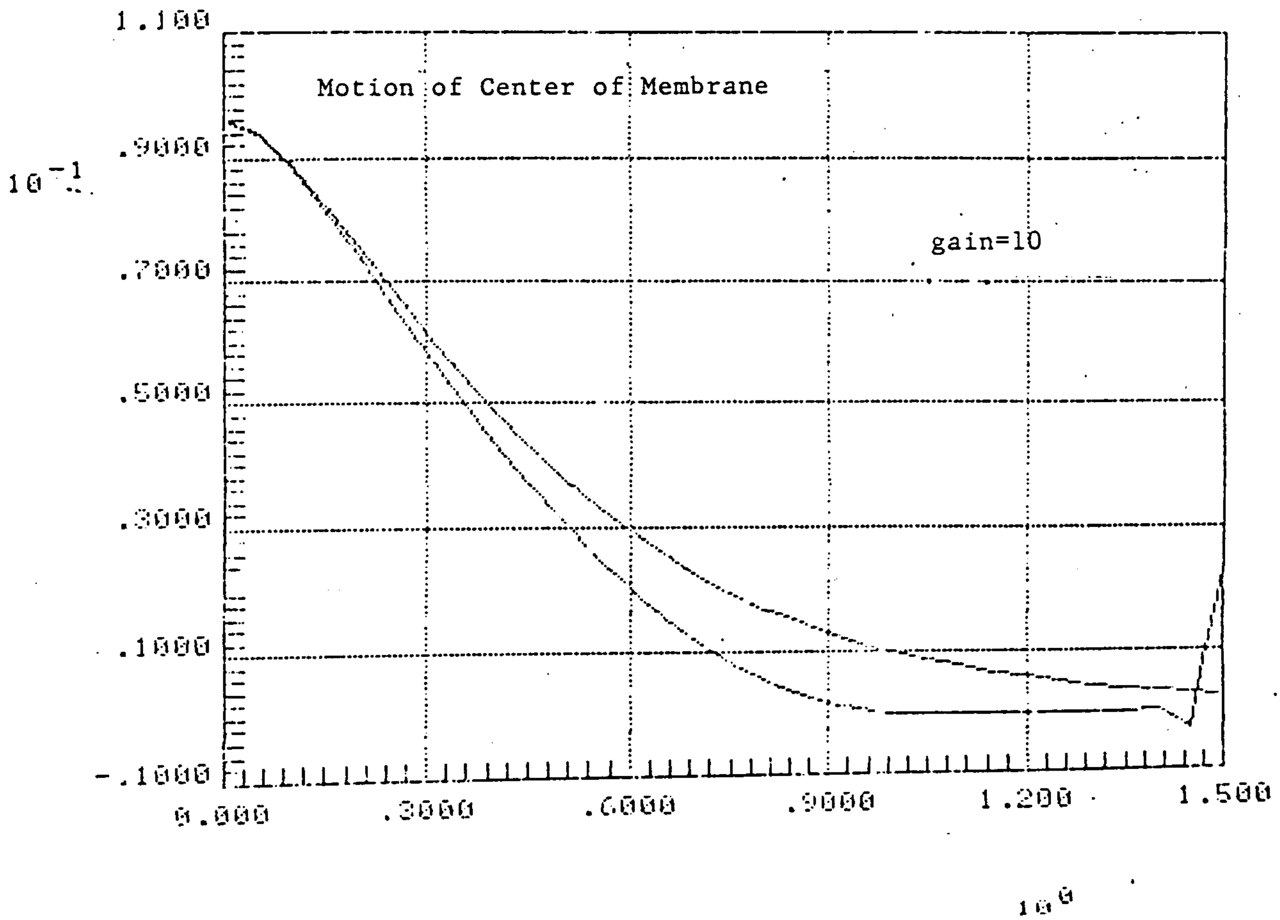


Fig. 19a: Numerical Simulation of the Motion of the Free Membrane Compared to Its Motion Under the Action of the Controller With Increasing Energy

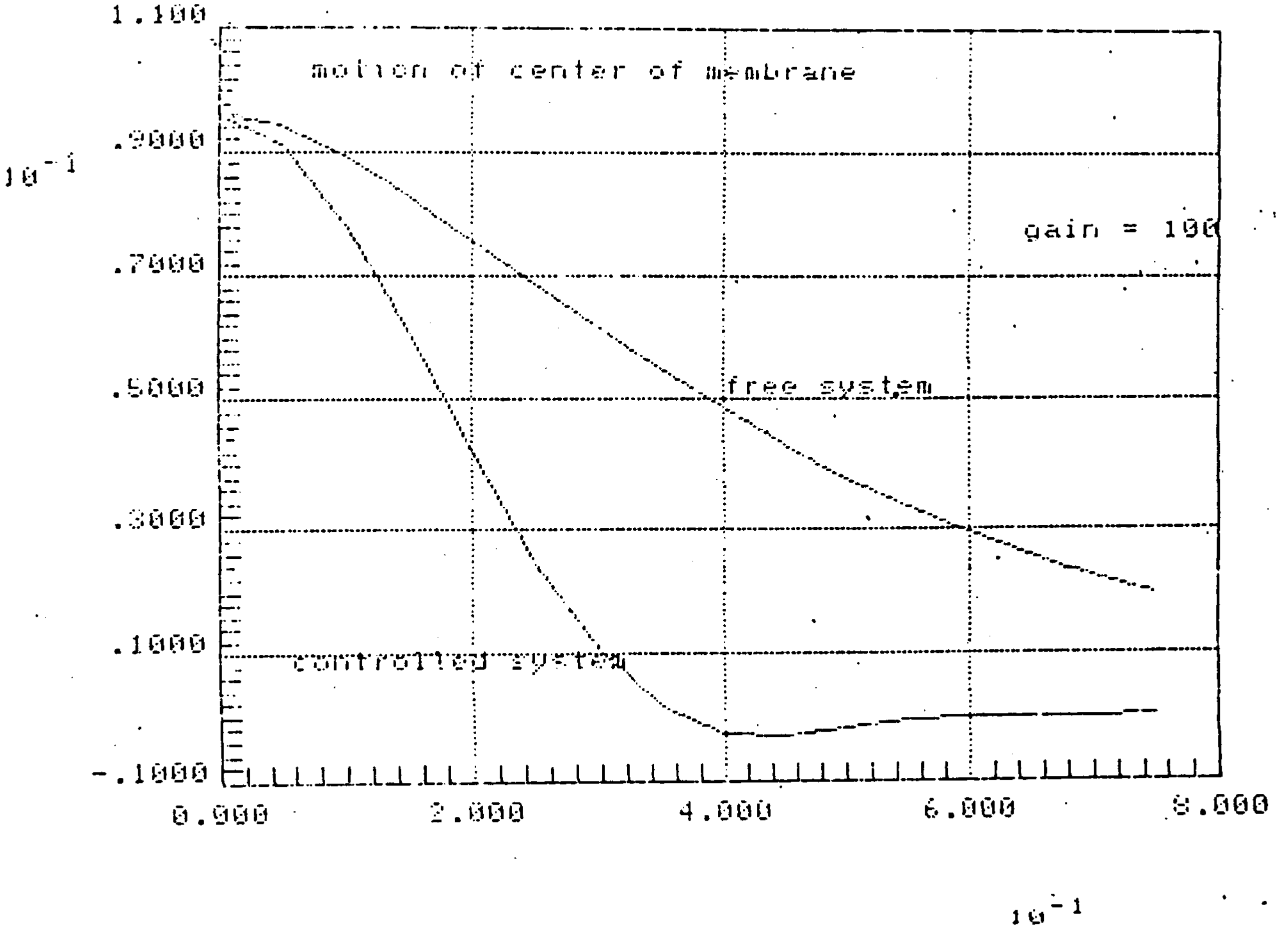


Fig. 19b: Numerical Simulation of the Motion of the Free Membrane Compared to Its Motion Under the Action of the Controller With Increasing Energy

2.4.8 Comparison Between Different Formulations of the Control Problem

(1) Multidimensional Control-Distributed Observation

Let us solve the following problem:

$$\begin{cases} \frac{\partial}{\partial t} \cdot \begin{bmatrix} h \\ h_t \end{bmatrix} = A \begin{bmatrix} h \\ h_t \end{bmatrix} + \sum_{k,\ell=1}^{N} u_{k\ell}(t) \begin{bmatrix} 0 \\ b_{k\ell}(x,y) \end{bmatrix} \\ y = c \begin{bmatrix} h \\ h_t \end{bmatrix} = \begin{bmatrix} K_1 h \\ K_2 h_t \end{bmatrix} \end{cases} \tag{2.283}$$

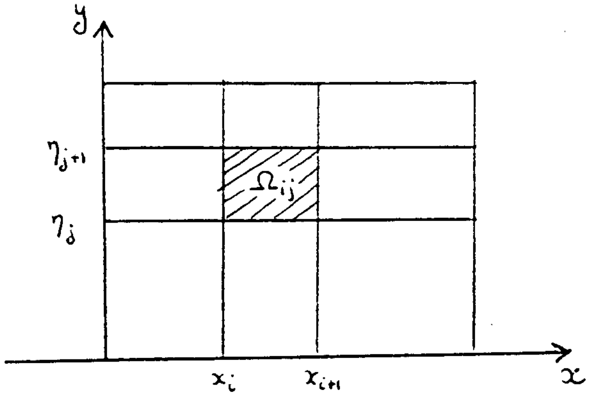


Fig. 20: Area of Application of Controller

and $b_{ij} = \begin{cases} 1 & (x,y) \in \Omega_{ij} \\ 0 & \text{elsewhere} \end{cases}$

Thus in this setting $G(jw)$ is an operator from $\mathbb{C}^{N^2} \to L^2 \oplus L^2$ and $G^*(j\omega)$ is the dual operator from $L^2 \oplus L^2 \to \mathbb{C}^{N^2}$. Therefore $G^*G(j\omega)$ is a linear operator from $\mathbb{C}^{N^2} \to \mathbb{C}^{N^2}$. Using the canonical isomorphism between finite dimensional vector spaces of the same dimension, G^*G can be regarded as a 4th order tensor in an N dimensional complex space ($\mathbb{C}^N$). Let $(v_{ijk\ell})$ be the components of the tensor associated to G^*G. We have shown (although not reported here for convenience) that:

$$v_{ijk\ell} = (K_1^2 + \omega^2 K_2^2) \iint_{\Omega} G_{ij}(x,y,\omega)\bar{G}_{k\ell}(x,y,\omega)dxdy \tag{2.284}$$

where $G_{ij}(x,y,\omega)$ is the integrated Green's function as usual.

(2) Point Controls:

In this case we assume that the input map is given by:

$$B : \begin{bmatrix} u_{11}(t) \\ u_{12}(t) \\ \vdots \\ u_{NN}(t) \end{bmatrix} \in \overset{N^2}{\mathbb{R}} \to \sum_{k=1}^{N} \sum_{\ell=1}^{N} u_{k\ell}(t) \begin{bmatrix} 0 \\ \delta(x - \zeta_k, y - \eta_\ell) \end{bmatrix} \tag{2.285}$$

i.e. we assume that we have point controllers located at the points (ζ_k, η_ℓ). We can show that the tensor $v_{ijk\ell}$ representing $G^*G(j\omega)$ is given by:

$$v_{ijk\ell} = (K_1^2 + K_2^2\omega^2) \iint_\Omega \bar{G}(x, y, \zeta_k, \eta_\ell, \omega) G(x, y, \zeta_i, \eta_j, \omega) dx dy \tag{2.286}$$

(The proof is omitted for convenience).

Hence we get the same result as in the case of piecewise constant control. This should not come as a surprise since the Green's function itself is approximated by piecewise constant function over small rectangle Ω_{ij}; therefore the system cannot "distinguish" between a point control located in the middle of Ω_{ij} and a piecewise constant control over Ω_{ij}.

(3) Point Control - Point Observation

Now instead of observing the whole membrane (distributed observation) we observe only a discrete number of points located on the membrane, i.e.

$$c : \begin{bmatrix} h \\ h_t \end{bmatrix} \in L^2 \oplus L^2 \to \begin{bmatrix} h(x_1, y_1, t) \\ \vdots \\ h(x_N, y_N, t) \end{bmatrix} \tag{2.287}$$

Note that $h(x_i, y_j, t) = \iint_\Omega h(x, y, t)\delta(x - x_i, y - y_j) dx dy$ and that C is an unbounded output operator.

Again we report only the end result:

$$v_{ijk\ell} = \sum_{m=1}^{N} \sum_{n=1}^{N} \bar{G}(x_m, y_n, \zeta_k, y_\ell, \omega) G(x_m, y_n, \zeta_i, \eta_j, w) \tag{2.288}$$

In other words assuming point observations amounts to approximating the double integral $\iint\limits_{\Omega} \bar{G}G$ by a 1st order approximation $\sum_m \sum_n \bar{G}G$. Since the double sum converges to the double integral as the mesh sizes go to 0 we conclude that the point observation approach is legitimate.

Theorem 2.18:

Consider the distributed observation problem (2.283) for the membrane and let

$$G(j\omega) = CR(j\omega; A)B \tag{2.289}$$

be the transfer function of the system and $(v_{ijk\ell})$ be the tensor corresponding to the operator G^*G as given by expression (2.284).

Consider also the point-observation problem (2.287) and let $G_N(j\omega)$ be its transfer function, and $(v^N_{ijk\ell})$ be the tensor corresponding to the operator $G^*_N G_N$ as given by expression (2.288). Then $(v^N_{ijk\ell})$ converges uniformly to $(v_{ijk\ell})$ as $N \to \infty$.

CHAPTER 3

THE STRING PROBLEM

In this chapter we solve the optimal state feedback control problem for the string. This model is very useful since it is the simplest hyperbolic system that is a good representation of the vibrations of the physical string when the effect of bending rigidity is small. Furthermore this relatively simple model brings out many interesting problems encountered in controlling distributed systems. It can also be helpful in giving insight into more complicated problems such as:

- the membrane problem.
- the beam problem. . .etc.

We begin our study by using an eigenfunction expansion technique for the simulation of the forward system and use a change of variables to handle the viscous damping, we present some examples.

The existence of uniform damping γ is imposed by physical considerations as any physical system dissipates energy. We will see later how to introduce better models for damping. From the mathematical point of view:

- it shifts the poles of the system away from the imaginary axis; the gain formulae involving contour integrals along the imaginary axis are well defined only if the singularities are shifted.
- the damping also guarantees the stabilizability of the system to insure the existence of optimal control (along with other conditions). One advantage is that it allows sidestepping the controllability question which involves solving moment problems in Hilbert space.

On the other hand the existence of damping makes it more difficult to devise exact formulae in the form of series expansion, and introduces exponential terms $e^{-\gamma t}$ to be handled with care not to destabilize the solution.

We then proceed to the computation of the Green's function by standard techniques and give an example.

In section 3.3 we quote some results from Russell's work [60] on the study of the controllability question for the string and give a physical interpretation.

We then go on to study the problem of localized optimal control of the string and give different alternatives, while trying to relate this case to the more general control problems treated in the following sections. The flexibility of these different approaches for the designer is discussed.

Next we study the convergence of step controls to point control which stems from physical considerations. We relate it to the approximation of identity in Harmonic Analysis. Although we do not proceed to test this convergence analysis numerically we give all the relevant formulae. The reason we stop at this level of the analysis is because we have already implemented the optimal point control with success.

The next section is the natural extension of the previous one and deals with the problems of multidimensional discrete control-discrete observation. We then proceed to solve the distributed control problem and show the multidimensional discrete problem studied earlier is a first order approximation. We also see that it leads to "approximating" the operator spectral factorization by a matrix spectral factorization.

The next step is to compute the control along a mode of the string (shape function) introducing a great simplicification in the computations. This is no surprise since the eigenfunction constitute a computational tool tailored to the particular system of interest. The scalar case serves as a preparation to the multidimensional case.

The next section deals with the problem of computing the optimal control in a basis of N eigenfunctions which leads to N decoupled scalar control problems and constitutes a very important improvement over the methods presented earlier.

We then study the question of finding approximations to the transfer function when the Green's function is not known (or difficult to get).

The last section deals with the point control by using numerical inverse Laplace transform which deals with the "δ-function" in integrated form and avoids the difficulty of dealing with a distribution directly.

3.1 Eigenfunction Expansion of the String Equation

Consider the string equation:

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} = \alpha \frac{\partial^2 h}{\partial x^2} + \gamma h + \beta \frac{\partial h}{\partial t} + u & x \epsilon (0, \ell) \\ h(0) = h(\ell) = 0 & \text{B.C.} \\ h(0, x) = h_0(x) & \\ h'(0, x) = h'_0(x) & \text{I.C.} \end{cases} \tag{3.1}$$

First of all we introduce a change of dependent variable to eliminate the damping term:

$$g(t) = h(t) e^{-\frac{\beta}{2}t}$$

$$w(t) = u(t) e^{-\frac{\beta}{2}t} \tag{3.2}$$

which leads after reductions to:

$$\begin{cases} \frac{\partial^2 g}{\partial t^2} = \alpha \frac{\partial^2 g}{\partial x^2} + (\gamma + \frac{\beta^2}{4}) g + w & \\ g(0) = g(\ell) = 0 & \text{B.C.} \\ g(0) = h(0) & \\ g'(0) = h'(0) - \frac{\beta}{2} h(0) & \text{I.C.} \end{cases} \tag{3.3}$$

The associated eigenvalue problem is:

$$\begin{cases} \alpha \frac{\partial^2 \varphi_n}{\partial x^2} + \delta \varphi_n = \lambda_n \varphi_n \\ \varphi_n(0) = \varphi_n(\ell) = 0 \end{cases} \tag{3.4}$$

where

$$\delta = \gamma + \frac{\beta^2}{4} \tag{3.5}$$

which is easily seen to admit the solution:

$$\varphi_n(x) = \sin(\frac{n\pi}{\ell}x) \tag{3.6}$$

with the corresponding eigenvalues

$$\lambda_n = (\gamma + \frac{\beta^2}{4}) - \alpha(\frac{n\pi}{\ell})^2 \tag{3.7}$$

The associated initial value problem is

$$\left\{ \begin{array}{l} \frac{d}{dt}\begin{bmatrix} a_n(t) \\ \dot{a}_n(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \lambda_n & 0 \end{bmatrix} \begin{bmatrix} a_n(t) \\ \dot{a}_n(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w_n(t) \\ \begin{bmatrix} a_n(0) \\ \dot{a}_n(0) \end{bmatrix} = \begin{bmatrix} r_n \\ v_n \end{bmatrix} \end{array} \right. \tag{3.8}$$

Let $\mu_n = \sqrt{|\lambda_n|}$ then the solution will be:

$\underline{\lambda_n < 0}$:

$$\left\{ \begin{array}{ll} a_n(t) & = r_n \cos \mu_n t + \frac{v_n}{\mu_n} \sin \mu_n t + \frac{1}{\mu_n} \int_o^t w_n(\sigma) \sin \mu_n(t-\sigma) d\sigma \\ \dot{a}_n(t) & = -\mu_n r_n \sin \mu_n t + v_n \cos \mu_n t + \int_o^t \cos \mu_n(t-\sigma) w_n(\sigma) d\sigma \end{array} \right. \tag{3.9}$$

$\underline{\lambda_n > 0}$:

$$\left\{ \begin{array}{ll} a_n(t) & = r_n ch\mu_n t + \frac{v_n}{\mu_n} sh\mu_n t + \int_0^t w_n(\sigma) sh\mu_n(t-\sigma) d\sigma \\ \dot{a}_n(t) & = r_n \mu_n sh\mu_n t + v_n ch\mu_n t + \int_0^t w_n(\sigma) ch\mu_n(t-\sigma) d\sigma \end{array} \right. \tag{3.10}$$

$\underline{\lambda_n = 0}$:

$$\left\{ \begin{array}{ll} a_n(t) & = r_n + v_n t + \int_o^t (t-\sigma) w_n(\sigma) d\sigma \\ \dot{a}_n(t) & = v_n + \int_o^t w_n(\sigma) d\sigma \end{array} \right. \tag{3.11}$$

This leads to the solution of the BVP in the form:

$$g(x,t) = \sum_{n=1}^{\infty} a_n(t) \varphi_n(x) \tag{3.12}$$

Finally the displacement and velocity of the string are given by:

$$h(x,t) = e^{\frac{\beta}{2}t} \sum_{n=1}^{\infty} a_n(t)\varphi_n(x)$$

$$h_t(x,t) = \sum_{n=1}^{\infty}(\dot{a}_n(t) + \frac{\beta}{2}a_n(t))e^{\frac{\beta}{2}t}\varphi_n(x) \tag{3.13}$$

<u>EXAMPLE</u>:

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} = \alpha\frac{\partial^2 h}{\partial x^2} + \beta\frac{\partial h}{\partial t} + \gamma h + u \\ h(0) = h(\ell) = 0 \\ h_0(x) = a\varphi_n(x) \\ h_0'(x) = (b+d)\varphi_n(x) \end{cases} \tag{3.14}$$

where the control is given by:

$$u(t) = a(1+\sigma-\gamma)\cos t + b(1+\sigma+\beta-\gamma)e^t + d(1+\sigma-\gamma) - \beta c + a\beta\sin t$$

and $\sigma = \alpha(\frac{n\pi}{\ell})^2, \varphi_n(x)$ being the i^{th} eigenfunction.

The displacement and velocity are then given by:

$$h(t,x) = (a + bt + c\cos t + de^t)\varphi_n(x)$$

$$h_t(x,t) = (b - c\sin t + de^t)\varphi_n(x) \tag{3.15}$$

This example serves as a test for the simulation of the system, combining constant; harmonic and exponential terms.

3.2 Green's Function in the Frequency Domain

Using Laplace transform, the string equation:

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} = \alpha\frac{\partial^2 h}{\partial x^2} + \beta\frac{\partial h}{\partial t} + \gamma h + u \\ \text{B.C.} \\ \text{I.C.} \end{cases} \tag{3.16}$$

becomes in the frequency domain:

$$\frac{\partial^2 H}{\partial x^2} - \mu^2(s)H = V(s,x) \tag{3.17}$$

where $\mu^2(s)$ is a polynomial of degree 2:

$$\mu^2(s) = \mu_0 + \mu_1 s + \mu_2 s^2 \tag{3.18}$$

To get the Green's function we need to solve:

$$\begin{cases} \frac{\partial^2 G}{\partial x^2}(s, x|x') - \mu^2(s) G(s, x/x') = \delta(x - x') \\ G|_{\partial\Omega} = 0 \end{cases} \tag{3.19}$$

It can be checked that the solution is:

$$G(s, x/x') = -\frac{1}{\mu(s) sh(\mu(s)\ell)} \begin{cases} sh(\mu(s)(\ell - x') sh\mu(s)x & 0 \leq x \leq x' \\ sh(\mu(s)x') sh(\mu(s)(\ell - x)) & x' \leq x \leq \ell \end{cases} \tag{3.20}$$

The solution of the general system will be given in terms of the Green's function as follows:

$$H(x, s) = \int_0^\ell G(s, x \mid x') V(s, x') dx' \tag{3.21}$$

EXAMPLE:

To test our integration scheme we consider the following forcing term:

$$V(x, \omega) = \{\mu^2(\omega) - (\frac{n\pi}{\ell})^2\}(\alpha + \beta\omega + \gamma \cos\omega + \lambda e^\omega)\varphi_n(x) \tag{3.22}$$

which generates the response:

$$H(x, \omega) = (\alpha + \beta\omega + \gamma \cos\omega + \cos\omega + \lambda e^\omega)\varphi_n(x) \tag{3.23}$$

3.3 The Controllability Question

As a preparation for the study of the optimal control of the string equation one needs to investigate the controllability question of the system. In his paper [60] Russell Shows that:

- the problem reduces to a moment problem in a Hilbert space.
- physically the control time has to be greater than a certain fixed value determined by the finite speed of the wave. This is in contrast with the finite dimensional case where instantaneous control is possible.

Let us summarize Russell's results. Consider the equation:

$$\begin{cases} \rho(x)\frac{\partial^2 u}{\partial t^2} - \frac{\partial}{\partial x}(\rho(x)\frac{\partial u}{\partial x} = g(x)f(t) \\ \text{B.C.} \\ \text{I.C.} \end{cases} \tag{3.24}$$

A separation of variables:

$$u(x,t) = \alpha(t)\varphi(x) \tag{3.25}$$

leads to a Sturm-Liouville problem:

$$\frac{\partial^2 \varphi}{\partial x^2}(x) + \Gamma(x)\varphi(x) + \lambda\varphi(x) = 0 \tag{3.26}$$

To every $\psi(x)\epsilon L_2[0,\ell]$ corresponds the eigenfunctions expansion:

$$\psi(x) = \sum_{k=0}^{\infty} \psi_k \varphi_k(x) \tag{3.27}$$

Now the solution is:

$$u(x,t) = \sum_{k=0}^{\infty} \beta_k(t)\varphi_k(x) \tag{3.28}$$

One is naturally led to the moment problem:

$$\begin{cases} \int_0^t \sin\omega_k t \cdot h(t)dt = c_k \\ \int_0^t \cos\omega_k t \cdot h(t)dt = d_k \end{cases} \tag{3.29}$$

In the general case the $\{\omega_k\}$ are not multiples of a fixed constant, and it is necessary to use nonharmonic Fourier series $\{e^{i\omega_k t}, e^{-i\omega_k t}; k = 0, 1, 2, \cdots\}$ to solve the above problem.

Define:

$$\tau = \int_0^1 \sqrt{\frac{\rho(x)}{p(x)}} dx \tag{3.30}$$

then the system is controllable in time τ when the condition $T > 2\tau$ holds.

The study of the controllability question is a nice preparation for our work (existence of control and physical insight). However it is not essential because our system contains damping (always present in physical systems) which guarantees the weaker condition of stabilizability. This condition (along with other hypotheses) is sufficient for the existence of optimal control.

3.4 Localized Control Problem

We are interested in computing the optimal control when the controller is locazlied at a point ζ of the string, with the observation at some other point η of the string (there is no difficulty in having $\eta = \xi$ *i.e.* control and observation localized at same point). However we take $\eta \neq \xi$ in general to give the designer more flexibility in localizing controls and sensors.

We first transform the string equation into a first order system

$$\left\{ \begin{array}{l} \frac{d}{dt} \begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \alpha \frac{\partial^2}{\partial x^2} + \gamma & \beta \end{bmatrix} \begin{bmatrix} h \\ h' \end{bmatrix} + u(t) \begin{bmatrix} 0 \\ \delta(x - \xi) \end{bmatrix} \\ y = kh(x - \eta) \\ \text{B.C.} \\ \text{I.C.} \end{array} \right. \tag{3.31}$$

The resolvent operator is defined by:

$$R(j\omega; A) = (j\omega - A)^{-1} \tag{3.32}$$

for $j\omega\epsilon\rho(A)$ resolvent set of A.

Given $(h, h')\epsilon L^2 \oplus L^2$ we seek $(\varphi, \psi)\epsilon L^2 \oplus L^2$ such that:

$$R(j\omega; A) \begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} \varphi \\ \psi \end{bmatrix} \tag{3.33}$$

Using the formal expression for $R(j\omega; A)$:

$$\begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} j\omega & -1 \\ -\alpha\partial_{xx-\gamma} & j\omega - \beta \end{bmatrix} \begin{bmatrix} \varphi \\ \psi \end{bmatrix} \tag{3.34}$$

which leads us to the following BVP:

$$\begin{cases} \alpha\frac{\partial^2\varphi}{\partial x^2} + (\omega^2 + \gamma + j\omega\beta)\varphi = (j\omega - \beta)h - h' \quad \text{in } \mathring{\Omega} \\ \varphi|_{\partial\Omega} \\ \psi = j\omega\varphi - h \\ \Omega = [0, \ell] \end{cases} \tag{3.35}$$

The functions h and h' appear in the forcing term as expected.

In operator notation the transfer function is given by:

$$G(j\omega) = CR(j\omega; A)\ B \tag{3.36}$$

The PDE characterizing the resolvent operator can be put in the simpler form:

$$\begin{cases} a\frac{\partial^2\varphi}{\partial x^2} - \mu^2(\omega)\varphi = v \\ \varphi|_{\partial\Omega} = 0 \end{cases} \tag{3.37}$$

then use the Green kernel to give an expression of the solution:

$$\varphi(x, \omega) = \int_0^\ell \ G(\omega, x \mid x')\ \ v(x', \omega)\ \ dx' \tag{3.38}$$

In our special case

$$Bu = u \begin{bmatrix} 0 \\ \delta(x-\xi) \end{bmatrix} \tag{3.39}$$

therefore

$$\begin{aligned} v &= -u\delta(x-\xi) \\ \varphi(x,\omega) &= -j\omega u G(\omega, x \mid \xi) \end{aligned} \tag{3.40}$$

Using the Observation equation for C gives:

$$G(j\omega) = -KG(\omega, \eta/\xi) \tag{3.41}$$

the adjoint is given by the conjugate expression

$$G^*(j\omega) = -K\bar{G}(\omega, \eta/\xi) \tag{3.42}$$

The spectral factorization algorithm will apply to:

$$F(j\omega) = 1 + G^*G(j\omega) = 1 + K^2 \mid G(\omega, \eta/\xi) \mid^2 \tag{3.43}$$

The general expression of the gains gives:

$$\begin{aligned} B^*K \begin{bmatrix} h \\ h' \end{bmatrix} = -\frac{K^2}{2\pi} \int_{-\infty}^{+\infty} & d\omega \ (F^-(j\omega))^{-1} \ \bar{G}(\omega, \eta/\xi) \\ & \int_0^\ell G(\omega, \eta/x')((j\omega - \beta)h(x') - h'(x'))dx' \end{aligned} \tag{3.44}$$

<u>Theorem 3.1</u>:

Consider the point control-point observation problem (3.31) for the string.

Define:

$$\begin{cases} g_1(\omega) = \bar{G}(w, \eta/\xi) \int_0^\ell G(\omega, \eta/x')h'(x')dx' \\ g_2(\omega) = \bar{G}(\omega, \eta/\xi) \int_0^\ell G(\omega, \eta/x')h'(x')dx' \end{cases} \tag{3.45}$$

where $g_1(\omega)$ is a function of the displacement alone whereas $g_2(\omega)$ depends on the velocity only.

Let the gain kernel be:

$$k(\omega) = (j\omega - \beta)g_1(\omega) - g_2(\omega) \tag{3.46}$$

Then the optimal state feedback control is given by:

$$u = -B^{*}K \begin{bmatrix} h \\ h' \end{bmatrix} = \frac{K^2}{2\pi} \int_{-\infty}^{+\infty} \; k(\omega)\, (F^{-}(j\omega)^{-1} d\omega \tag{3.47}$$

<u>Remark</u>:

There is a slightly more general situation to be considered, but more complicated from the programming point of view. It considers the case of a string controlled at one point only but observed at different points along the string.

One can proceed from this configuration allowing for discrete observation points which amounts to first order approximations of the integrals involving the Green kernel and different functions of displacement and velocity.

The other alternative would be to reconstruct the state of the system from the discrete observation and have more flexibility for the approximation of the integrals. This latter approach has the advantage of allowing arbitrary accuracy to be achieved using adaptive integration schemes based on spline polynomials of arbitrary degree. However one has to estimate the state in this case which is by no means a trivial problem.

Putting these considerations aside for the time being, we give the relevant formulae that can be derived by the usual techniques. The observation equation is changed to yield:

$$c \begin{bmatrix} h(\cdot) \\ h'(\cdot) \end{bmatrix} = k \; h(\cdot) \tag{3.48}$$

The "transfer function" operator becomes:

$$G(j\omega) \cdot u = -KuG(\omega, \cdot/\xi) \tag{3.49}$$

Similarly the adjoint operator becomes:

$$G^*(j\omega)\varphi = -K \int_0^\ell \bar{G}(\omega, x'/\xi)\varphi(x') \ dx' \tag{3.50}$$

The function to be spectral-factorized is:

$$F(\omega) = 1 + K^2 \int_0^\ell \ | G(\omega, x'/\xi) |^2 \ dx' \tag{3.51}$$

The optimal gains become:

$$B^*K \begin{bmatrix} h \\ h' \end{bmatrix} = -\frac{K^2}{2\pi} \int_{-\infty}^{+\infty} d\omega \ (F^-(j\omega))^{-1}$$

$$\int_0^\ell dx' G(\omega, x'/\xi)$$

$$\int_0^\ell G(u, x'/x'')((j\omega - \beta)h(x'') - h'(x''))dx'' \tag{3.52}$$

If we were to allow for multidimensional control and scalar observation instead, the situation will be the exact dual of the present case, but this does not seem to have any practical application.

We will say more when we cover the much more general case (but surprisingly the easiest for computations) when we control the whole string while observing the displacement of the string. The numerical implementation of this approach is within the framework of our method of computing optimal control of distributed systems which was successfully implemented for the membrane equation. In fact the string equation treated here is the analog of the membrane equation with one less dimension.

This is why it is interesting to test different approaches for optimal control of the string equation and infer the same conclusions for the membrane for which the technical mathematics are more complex. This is very helpful:

- from the <u>theoretical</u> point of view one can gain better insight by considering different approaches and how they relate. It can also serve as a tool for proving <u>convergence</u> of discrete methods to the case of full information.

- from the practical point of view this is really a precious tool for design by being free of practical restrictions like having to have concatenated controllers and sensors, or a large number of sensors to fit a fixed rigid theoretical formulation.

3.5 Approximation of the Identity and Convergence to a Point Control

In this section we consider the convergence question when we are approximating the Dirac "δ-function" by L^2-functions. This serves two purposes:

- in practice the physical realization will certainly involve a control distributed on a finite area
- it is also related to the eigenfunction expansion of the control.

The problem of approximation of the identity has been studied by Harmonic Analysts: approximations to the "δ-functions" serve as an approximate unity in the Banach Algebra $L^1(\mathcal{R})$ with convolution as multiplication.

Let us return to the point control of the previous section:

$$\begin{cases} \frac{d}{dt}\begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \alpha\frac{\partial^2}{\partial x^2} + \gamma & \beta \end{bmatrix}\begin{bmatrix} h \\ h' \end{bmatrix} + u(t)\begin{bmatrix} 0 \\ \delta(x-\xi) \end{bmatrix} \\ y = Kh(t,\eta) \\ \text{B.C.} \\ \text{I.C.} \end{cases} \tag{3.53}$$

The solution of this problem will be defined in a weak sense only because the forcing term is a distribution.

To define the convergence question more precisely let us consider the following sequence of simple functions:

$$f_h(x) = \begin{cases} \frac{1}{h} & \xi - \frac{h}{2} < x < \xi + \frac{h}{2} \\ 0 & \text{otherwise} \end{cases} \tag{3.54}$$

The sequence $f_h(x)$ converges in distributional sense to $\delta(x-\xi)$:

$$f_h(x) \quad \overset{h\to 0}{\rightharpoonup} \quad \delta(x-\xi) \tag{3.55}$$

Does it hold that the sequence of controls (corresponding to forcing terms $u_h(t)f_h(x)$ converges to the solution $u(t)$ (when the forcing term is $u(t)\delta(x-\xi)$?

From Harmonic Analysis we know that given a kernel $k(x)\varepsilon L^1(\Omega)$ and given

$$k_\varepsilon(x) = \frac{1}{\varepsilon}k(\frac{x}{\varepsilon}). \tag{3.56}$$

if $f\epsilon L^p(\Omega) \quad 1 \leq p < \infty$ then:

$$f * k_\varepsilon \to f \quad \text{as } \varepsilon \to 0 \quad \text{in} \quad L^p(\Omega) \tag{3.57}$$

If we take a smooth kernel $k\epsilon C^\infty(\Omega)$ and consider the sequence $f * k_\varepsilon$ the convergence is going to be pointwise under appropriate conditions on k.

However the sequence we have chosen $f_h(x)$ makes the computations easier and would be more meaningful physically too.

Next we consider distributions defined as linear functionals on function spaces. The inner product serves to define the derivative of a distribution by "transposing" the differential operator to the C^∞-test function:

$$< \partial_i\mu, \varphi >= - < \mu, \partial_i\varphi > \quad \text{(integration by parts)}$$

or

$$\int \varphi \cdot \partial_i u = \int \mu \cdot \partial_i\varphi \quad (\partial_i = \frac{\partial}{\partial x_i}) \tag{3.58}$$

We can give the following examples:

(1) $H(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases}$

$\partial H = \delta$

(2) $(fH)' = f'H + f(o)\delta$

Now we study the response of the string to a point control, and we consider the following discontinuous displacement function and compute the control giving rise to such a response:

$$h(t,x) = h(t)\psi(x) \tag{3.59}$$

with

$$\psi(x) = \begin{cases} \frac{x}{\xi} & 0 \le x \le \xi \\ \frac{x-\ell}{\xi-\ell} & \xi \le x \le \ell \end{cases} \tag{3.60}$$

The derivative of ψ is:

$$\psi'(x) = \begin{cases} \frac{1}{\xi} & 0 \le x < \xi \\ \frac{1}{\xi-\ell} & \xi < x \le \ell \end{cases} \tag{3.61}$$

The derivative of $\psi'(x)$ is:

$$\psi''(x) = \frac{1}{\xi(\xi-\ell)} \cdot \delta(x-\xi) \tag{3.62}$$

From the assumed expression (3.59) and the string equation we get:

$$u(x,t) = (h''(t) + \beta h'(t))\psi(x) - \frac{\alpha h(t)}{\xi(\xi-\ell)}\delta(x-\xi) \tag{3.63}$$

For convenience let us choose $h(t) = -\frac{1}{\beta}e^{-\beta t}$ so that:

$$h(x,t) = -\frac{1}{\beta}e^{-\beta t}\psi(x)$$

and

$$u(x,t) = \frac{\alpha e^{-\beta t}}{\beta\xi(\xi-\ell)}\delta(x-\xi) \tag{3.64}$$

We have that as $\epsilon \to 0$:

$$u_\epsilon(x,t) = \frac{\alpha e^{-\beta t}}{\beta\xi(\xi-\ell)}f_\epsilon(x) \rightharpoonup u(x,t) \tag{3.65}$$

we would like to check that:

$$h_\epsilon(x,t) \to h(x,t) = -\frac{1}{\beta}e^{-\beta t}\psi(x) \tag{3.66}$$

Now we need to compute the Fourier Series of the impulse control and we consider the asymptotic expansion of $f_\epsilon(x)$:

$$f_\epsilon(x) = \sum_{n=1}^{\infty} f_n(\epsilon) \cdot \sin \frac{n\pi}{\ell} x \tag{3.67}$$

Using the identities:

$$\begin{cases} \int_0^\pi \quad \sin my \cdot \sin ny \quad dy = 0 \quad m \neq n \\ \int_0^\pi \quad \sin^2 ny \quad dy = \frac{\pi}{2} \end{cases} \tag{3.68}$$

We get $f_n = \frac{\ell}{2\epsilon} \int_{\xi-\frac{\ell}{2}}^{\xi+\frac{\ell}{2}} \sin \frac{n\pi}{\ell} x \quad dx$

After integration we get:

$$f_n = \frac{\ell^2}{\pi n \epsilon} \sin(\frac{n\pi}{\ell}\xi) \sin(\frac{n\pi\epsilon}{\ell}) \tag{3.69}$$

This leads to

$$f_\epsilon(x) = \sum_{n=1}^{\infty} \frac{\ell^2}{\pi n \epsilon} \sin(\frac{n\pi}{\ell}\xi) \sin(\frac{n\pi x}{\ell}) \sin(\frac{n\pi\epsilon}{2\ell}) \tag{3.70}$$

Therefore

$$u_\epsilon(x,t) = \sum_{n=1}^{\infty} \frac{\alpha e^{-\beta t}}{\beta\xi(\xi-\ell)} \frac{\ell^2}{\pi n \epsilon} \sin(\frac{n\pi}{\ell}\xi) \sin(\frac{n\pi}{\ell}x) \sin(\frac{n\pi\epsilon}{2\ell}) \tag{3.71}$$

In this section we have given all relevant formulae for the investigation of numerical convergence when the point control is approximated by L^2-function $f_\epsilon(x)$ which is imposed by:

- theoretical considerations: the Davis [22] formulation of the optimal feedback control allows for bounded input maps only.

- physical considerations: a controller is always distributed over a finite area.

3.6 Multi-Dimensional Point Control-Point Observation

Here we investigate the multi-dimensional problem allowing for controls to be applied over discrete points along the string while observing the displacement of the string at discrete points. As before we allow the control points and the observation points to be distinct for a greater latitude in design.

The problem studied here will approximate a distributed control problem as the number of points gets larger and larger. In fact it turns out that it is a first order approximation to the distributed problems.

Consider the following:

$$\frac{d}{dt}\begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ a\frac{\partial^2}{\partial x^2} + d & c \end{bmatrix}\begin{bmatrix} h \\ h' \end{bmatrix} + \sum_{i=1}^{N} u_i(t)\begin{bmatrix} 0 \\ \delta(x-\xi_i) \end{bmatrix}$$

$$y = Cx = k\begin{bmatrix} h(\eta_1,t) \\ h(\eta_2,t) \\ \vdots \\ h(\eta_N,t) \end{bmatrix} \tag{3.72}$$

Let the Resolvent operator be:

$$R(j\omega;a)\begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} \varphi \\ \psi \end{bmatrix} \tag{3.73}$$

we have seen before that the resolvent characterizing BVP is:

$$\begin{cases} a\frac{\partial^2\varphi}{\partial x^2} + (d+\omega^2+jc\omega)\varphi = (j\omega - c)h - h' \\ \varphi|_{\partial\Omega} = 0 \\ \psi = j\omega\varphi - h \end{cases} \tag{3.74}$$

The transfer function of the system is given in operator notation:

$$G(j\omega) = C\ R(j\omega;A)B \tag{3.75}$$

Applying it to a vector control we get:

$$G(j\omega)\underline{u} = \sum_{i=1}^{N} u_i(t)\cdot C\ R(j\omega;A)\begin{bmatrix} 0 \\ \delta(x-\xi_i) \end{bmatrix} \tag{3.76}$$

Keeping the notation of the resolvent equation we get:

$$\varphi(x,\omega) = -\int_0^{\ell} G(\omega, x \mid x')\delta(x' - \xi_i)dx' = -G(\omega, x \mid \xi_i) \tag{3.77}$$

The transfer function will be given by

$$G(j\omega)\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} = -K\sum_{i=1}^{N} u_i(t)\begin{bmatrix} G(\omega,\eta_1 \mid \xi_i) \\ G(\omega,\eta_2 \mid \xi_i) \\ \vdots \\ G(\omega,\eta_N \mid \xi_i) \end{bmatrix} \tag{3.78}$$

Let $G^*(j\omega)$ be the adjoint operator $G^*(j\omega)$ with:

$$G^* = [G^*_{ij}]_{1\leq i,j\leq N} \tag{3.79}$$

Then the equation

$$G^*\underline{v} = \underline{w} \tag{3.80}$$

is equivalent to

$$w_i = -K\sum_{j=1}^{N}\overline{G}(\omega,\eta_j \mid \xi_i)v_j$$

or

$$G^*_{ij} = -K\overline{G}(w,\eta_j \mid \xi_i) \tag{3.81}$$

The spectral factorization algorithm will be applied to:

$$F(\omega) = I + G^*G(j\omega) \tag{3.82}$$

If we let $G^*G = [G^*G_{ij}]$ then

$$G^*G_{ij} = K^2\sum_{k=1}^{N}\overline{G}(\omega,\eta_k \mid \xi_i)\cdot G(\omega,\eta_k \mid \xi_j) \tag{3.83}$$

Henceforth

$$F_{ij} = \delta_{ij} + K^2\sum_{k=1}^{N}\overline{G}(\omega,\eta_k \mid \xi_i)\cdot G(\omega,\eta_k \mid \xi_j) \tag{3.84}$$

To compute the gains let us recall the general formula:

$$[B^*K]\begin{bmatrix} h \\ h' \end{bmatrix} = \frac{1}{2\pi}\int_{-\infty}^{+\infty}(F^-(\omega))^{-1}G^*CR(j\omega;A)\begin{bmatrix} h \\ h' \end{bmatrix}d\omega \tag{3.85}$$

From the resolvent equation we get:

$$\varphi(x,\omega) = \int_0^\ell \quad G(\omega, x \mid x')((j\omega-\beta)h(x') - h'(x'))dx' \tag{3.86}$$

and $\psi(x,\omega) = j\omega\varphi(x,\omega) - h(x)$

therefore

$$CR(j\omega;A)\begin{bmatrix} h \\ h' \end{bmatrix} = K\int_0^\ell \quad G(\omega, x \mid x')((j\omega-\beta)h(x') - h'(x'))\ dx \tag{3.87}$$

We see that the formulae for $\varphi(x,\omega)$ and $CR(j\omega;A)\begin{bmatrix} h \\ h' \end{bmatrix}$ involve the displacement and velocity functions $h(x)$ and $h'(x')$.

However we observe only at discrete points, thus the best approach is to devise an approximation scheme for these integrals involving only the observed discrete values. This is much more efficient than estimating the state first then plug in the estimate in the relevant formulae. Thus for a given subdivision of the interval $(0,\ell)$:

$$0 = x_1 < x_2 < \cdots < x_{N+1} = \ell \tag{3.88}$$

We can take for example the 1st order Euler approximation:

$$\int_0^\ell \quad f(x)\ \ dx = \sum_{i=1}^{N}(x_{i+1}-x_i)f(\xi_i) \quad \xi_i\epsilon[x_i, x_{i+1}] \tag{3.89}$$

Applying this to $\varphi(x,\omega)$ at the observation points gives

$$\varphi(\eta_j,\omega) = \int_0^\ell \quad G(\omega,\eta_j \mid x')H(\omega,x')\ dx'$$

$$\sim \frac{\ell}{N}\sum_{k=1}^{N} G(\omega,\eta_j \mid \xi_k)((j\omega-\beta)h(\xi_k) - h'(\xi_k)) \tag{3.90}$$

Define the following gain functions:

$$g_i(\omega) = -\frac{K^2\ell}{N}\sum_{j=1}^{N}\sum_{k=1}^{N} \overline{G}(\omega,\eta_j \mid \xi_i)G(\omega,\eta_j \mid \xi_k)((j\omega-\beta)h(\xi_k)-h'(\xi_k)) \quad (3.91)$$

If we define the auxiliary functions

$$\chi_{ik}(\omega) = \sum_{j=1}^{N}\overline{G}(\omega,\eta_j \mid \xi_i)G(\omega,\eta_j \mid \xi_k) \quad (3.92)$$

Then the gain functions become

$$g_i(\omega) = -\frac{\ell K^2}{N}\sum_{k=1}^{N}\chi_{ik}(\omega)((j\omega-\beta)h(\xi_k)-h'(\xi_k)) \quad (3.93)$$

In matrix notation

$$\underline{g}(\omega) = -\frac{\ell K^2}{N}[\chi(\omega)]((j\omega-\beta)\underline{h}-\underline{h}') \quad (3.94)$$

where $[x,(\omega)]$ is the matrix of general element $x_{ik}(\omega)$.

The optimal state feedback control is given by:

$$\underline{u} = -[B^*K]\begin{bmatrix} h \\ h' \end{bmatrix} = -\frac{\ell K^2}{2\pi N}\int_{-\infty}^{+\infty}(F^-(j\omega))^{-1}[\chi(\omega)]((j\omega-\beta)\underline{h}-\underline{h}')d\omega \quad (3.95)$$

This approach was taken to show:

- the consistency of the formulation when passing from scalar to multidimensional control and the possibility of implementing it within our framework of matrix spectral factorization.
- to relate to the result of the next section dealing with distributed control and show that what we have here is just a first order approximation. This can be used to prove the convergence to the distributed control problem

because the formulation in the present section is the one corresponding to a physical realization.

In the next sections we are going to take a much more powerful method using the eigenfunctions of the system as basis for the state space which introduces a huge saving in computations.

3.7 Distributed Control

Let us consider the string problem with no constraints on the control and with full information on the state.

$$\begin{cases} \frac{d}{dt}\begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ a\frac{\partial^2}{\partial x^2}+\gamma & \beta \end{bmatrix}\begin{bmatrix} h \\ h' \end{bmatrix} + \begin{bmatrix} 0 \\ u(x,t) \end{bmatrix} \\ y = Kh \\ \text{B.C.} \\ \text{I.C.} \end{cases} \tag{3.96}$$

For the resolvent equation we keep the same notations as before.

The transfer function "operator" is in this case:

$$G(jw)\cdot u(x,t) = -K\int_0^\ell G(\omega, x \mid x')u(x',t)\ \ dx' \tag{3.97}$$

Similar computation give the adjoint operator

$$G^*(j\omega)\cdot\varphi(x') = -K\int_0^\ell \overline{G}(\omega, x \mid x')\varphi(x)\ \ dx \tag{3.98}$$

Notice that in this case the formulae for G and G^* are symmetrical and defined by integral operators with conjugate kernels. This leads to

$$G^*G(jw)\cdot u(x,t) = -K^2\int_0^\ell dy\ \ u(y,t)\int_0^\ell G(\omega, z \mid y)\overline{G}(\omega, x \mid \xi)\ \ dz \tag{3.99}$$

Define the "transfer" kernel:

$$k(\omega, x \mid y) = \int_0^\ell G(\omega, z \mid y)\overline{G}(\omega, x \mid y)\ \ dz \tag{3.100}$$

Then

$$G^*G[u] = -K^2 \int_0^\ell k(\omega, x \mid y) u(y,t) \ dy \tag{3.101}$$

Theorem 3.2:

Consider the distributed control/distributed observation (3.96) for the string.

The "modified" transfer function is given by

$$F(\omega)[u] = u + G^*G[u] = u(x,t) + K^2 \int_0^\ell k(\omega, x \mid y) u(y,t) \ dy \tag{3.102}$$

where $k(\omega, x|y)$ is given by expression (3.100).

In the distributed control problem we have an operator spectral factorization problem. This is the main advantage of the Wiener-Hopf technique compared with the Ricealti operator approach is that the dimension of the spectral factorization problem is the dimension of the control space. When we considered finite dimensional controls we arrived to matrix spectral factorizations. As the dimension N of the controls goes to ∞, the matrix transfer function will approach the operator transfer function. Let us try to recover the multidimensional discrete case from the present one by postulating a control of the form:

$$u(x,t) = \sum_{i=1}^{N} u_i(t) \delta(x - \xi_i) \tag{3.103}$$

Then

$$\begin{aligned} G^*G[u] &= K^2 \int_0^\ell k(\omega, x \mid y) \delta(y - \xi_i) dy \\ &= K^2 k(\omega, x \mid \xi_i) \end{aligned} \tag{3.104}$$

By using the expression for the kernel k we finally get:

$$G^*G[u] = K^2 \sum_{i=1}^{N} u_i(t) \int_0^\ell G(\omega, z \mid \xi_i) \overline{G}(\omega, x \mid z) \ dz \tag{3.105}$$

Now since the control $u(x,t)$ is this particular form can be identified with $[u_1(t) = u(\xi,t), \cdots, u_N(t) = u(\xi_N,t)]$ the above expression becomes:

$$G^*G[u] = K^2 \sum_{i=1}^{N} u_i(t) \int_0^{\ell} G(\omega, z \mid \xi_i)\overline{G}(\omega, \xi_j \mid z) \ \ dz \tag{3.106}$$

And we get a matrix approximation of G^*G:

$$G^*G_{ij} = \int_0^{\ell} G(\omega, z \mid \xi_i)\overline{G}(\omega, \xi_j \mid z) \ \ dz \tag{3.107}$$

The final step consists in taking this matrix approximation to the operator G^*G and use a discrete sum as an approximation for the integral. This is possible because we now make the assumption that the obervation is made at discrete points $z = \eta_1, \eta_2 \cdots \eta_N$. Hence

$$G^*G_{ij} = \sum_{k=1}^{N} \overline{G}(\omega, \xi_j \mid \eta_k) \cdot G(\omega, \eta_k \mid \xi_i) \tag{3.108}$$

Let us note however that the scalar point control cannot be obtained as a special case of the present one.

In the next sections we are going to use eigenfunction expansions to approximate a given shape instead of using simple functions $\sum_{ui} \delta(x - \xi_i)$ (or more exactly $\sum u_i f_\epsilon(x - \xi_i)$).

3.8 Control Along a Mode of the System

Let $\varphi(x,y)$ be an eigenfunction of the system. We consider the following optimal control problem:

$$\begin{cases} \dot{x} = Ax + u(t) \begin{bmatrix} 0 \\ \varphi(x) \end{bmatrix} \\ y = Cx \end{cases}$$

$$B : \mathbb{C} \to L^2 \oplus L^2 \quad Bu = \begin{bmatrix} 0 \\ u \cdot \varphi \end{bmatrix}$$

$$C : L^2 \oplus L^2 \to L^2 \quad Cx = h \ \ , \ \ x = \begin{bmatrix} h \\ \dot{h} \end{bmatrix} \tag{3.109}$$

We seek the optimal control that minimizes the quadratic criterion:

$$\int_0^\infty (\| u \|^2 + \| y \|^2) \ dt \tag{3.110}$$

Since the problem has been treated in detail for the membrane we just give the relevant formulae and refer to the previous chapter for details. In fact this is one advantage of our metholodogy which permits great flexibility by allowing the handling of this new problem by:

- replacing the old eigenfunctions by the new ones.
- feeding the new eigenvalues to the system.

In this particular case it really amounts to discarding one space dimension to get the string equation from the membrane equation.

The solution to the resolvent equation is:

$$f = \frac{\alpha(c - j\omega) - \beta}{\lambda + (d + \omega^2) + jc\omega} \cdot \varphi \tag{3.111}$$

with $\lambda = -a(\frac{n\pi}{\ell})^2$ (we had a different formulae for the membrane, and still another formulae for the beam) and

$$h = \alpha\varphi \ \ h' = \beta\varphi \tag{3.112}$$

The transfer function is:

$$G(j\omega)u = -\frac{Ku}{\lambda + (d + \omega^2) + jc\omega} \cdot \varphi \tag{3.113}$$

And therefore we have:

$$F(\omega) = 1 + \frac{K^2}{(\lambda + d + \omega^2) + c^2\omega^2} \int_\Omega \varphi^2 \tag{3.114}$$

Theorem 3.3:

The optimal state feedback gain for the mode shape control (3.108) of the string is given by:

$$B^*K \begin{bmatrix} h \\ h' \end{bmatrix} = -\frac{K^2}{2\pi} \cdot \int_\Omega \varphi^2 \int_{-\infty}^{+\infty} (F^-(w))^{-1} \frac{\alpha(c - jw) - \beta}{(\lambda + d + \omega^2)^2 + c^2\omega^2} \; d\omega \quad (3.115)$$

where $F(\omega)$ is given by expression (3.114), and $\alpha = \alpha(t) \quad \beta = \beta(t)$ are given by expression (3.112); the optimal control is:

$$u^*(t) = -B^*K \begin{bmatrix} h \\ h' \end{bmatrix} \quad (3.116)$$

We have given the formulae just to make the point clear that the method is valid without major change for different models and one needs only to plug the eigenvalues and eigenfunctions in the "algorithm" to generate the optimal control. In fact the "algorithm" accepts as much information as it is given to it, and the more information available the more efficient. We shall say more about this later.

3.9 Multidimensional Control Along Modes of the System

In this section we will see the distributed control problem from a different angle by using N eigenfunctions of the system as basis of the control space (and the state space it turns out, since eigenspaces are A-invariant). We are to derive the exact formulae in the form of series expansion for this hyperbolic system with finite dimensional control set. When we remove physical considerations and let $N \to \infty$ the solution converges to the distributed optimal control. Furthermore, in this case, the matrix to be spectral-factorized diagonalizes and we are reduced to N decoupled scalar control problems. This section serves to prove another point: that the knowledge of the Green's function is not absolutely necessary and we can deal with the situation by devising approximations to the transfer function.

So let us consider the following problem:

$$\begin{cases} \frac{d}{dt}\begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \alpha\frac{\partial^2}{\partial x^2}+\gamma & \beta \end{bmatrix}\begin{bmatrix} h \\ h' \end{bmatrix} + \sum_{i=1}^{N} u_i(t)\begin{bmatrix} 0 \\ \varphi_i(x) \end{bmatrix} \\ y = Kh \\ \text{B.C.} \\ \text{I.C.} \end{cases} \tag{3.117}$$

$\varphi_1, \varphi_2 \cdots \varphi_N$ are the first eigenfunctions of the system.

In the final paragraph we will talk about the physical realization and the relationship to discrete controls.

The resolvent equation stays the same and we keep the same notation.

The transfer function is:

$$G(j\omega)\underline{u} = CR(j\omega; A)\sum_{i=1}^{N} u_i \begin{bmatrix} 0 \\ \varphi_i(x) \end{bmatrix} \tag{3.118}$$

Let

$$R(j\omega; A)\begin{bmatrix} 0 \\ \varphi_i(x) \end{bmatrix} = \begin{bmatrix} \chi_i(x) \\ k_i(x) \end{bmatrix} \tag{3.119}$$

The solution to the resolvent equation is:

$$\chi_i(x,\omega) = -\int_0^{\ell} G(\omega, x \mid x')\varphi_i(x')dx' \tag{3.120}$$

To get a solution independent of the knowledge of $G(\omega, x \mid x')$ we use the bilinear expansion of G:

$$G(\omega, x \mid x') \sim \sum_{n=1}^{\infty} \frac{\varphi_n(x)\varphi_n(x')}{\lambda_n - \lambda(\omega)} \tag{3.121}$$

λ_n being the eigenvalue corresponding to the eigenfunction φ_n.

Note:

We use the full bilinear approximation and do not truncate to the first N terms.

Using this bilinear expansion into the expression of $\chi_i(x, w)$ and using the orthogonality of the eigenfunctions we get:

$$\chi_i(x, \omega) = \frac{\varphi_i(x)}{\lambda_i - \lambda(\omega)} \tag{3.122}$$

We see that $\chi_i(x, \omega)$ depends only on φ_i, λ_i.

Going back to the expression of the transfer function:

$$G(j\omega)\underline{u} = K \sum_{i=1}^{N} u_i(t) \frac{\varphi_i(x)}{\lambda_i - \lambda(\omega)} \tag{3.123}$$

We see that G depends only on $\varphi_1 \cdots \varphi_N$ (so far we did not introduce any truncation).

To compute G^* we consider a function $v(x)$ and its Fourier coefficients:

$$v_i = \int_0^\ell v(x)\varphi_i(x) \;\; dx \tag{3.124}$$

If we assume that

$$G^*(j\omega) \cdot v = \underline{w} \tag{3.125}$$

Then

$$w_i = K \frac{\overline{v}_i}{\lambda_i - \overline{\lambda}(\omega)} \tag{3.126}$$

For G^*G we get:

$$G^*G(j\omega) = G^*\{K \sum_{j=1}^{N} u_j \frac{\varphi_j(x)}{\lambda_j - \overline{\lambda}(\omega)}\}$$

$$= K \sum_{j=1}^{N} \frac{u_j}{\lambda_j - \overline{\lambda}(\omega)} \cdot G^*\varphi_j(x) \tag{3.127}$$

Let $\underline{w}^j = G^*\varphi_j(x)$

Then

$$w_i^j = \frac{K}{\lambda_i - \overline{\lambda}(\omega)} \int_0^\ell \varphi_i(x)\varphi_j(x)dx = \delta_{ij}\frac{K}{\lambda_i - \overline{\lambda}(\omega)} \tag{3.128}$$

Therefore

$$w_i^j = \begin{cases} 0 & i \neq j \\ \frac{K}{\lambda_i - \overline{\lambda(\omega)}} & i = j \end{cases} \tag{3.129}$$

The expression for G^*G becomes:

$$G^*G(j\omega) \cdot \underline{u} = K^2 \sum_{j=1}^{N} \frac{u_j}{\mid \lambda j - \lambda(\omega) \mid^2} \vec{e}_j \tag{3.130}$$

where

$$\vec{e}_j \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \leftarrow j^{th} \text{ position} \tag{3.131}$$

The expression of $F(\omega)$ can be deduced from

$$F(\omega) = I + G^*G \tag{3.132}$$

and the above expression for G^*G

$$F(\omega) = \begin{bmatrix} 1 + \frac{k^2}{|\lambda_1 - \overline{\lambda}(\omega)|^2} & & 0 \\ & \ddots & \\ 0 & & 1 + \frac{k^2}{|\lambda_N - \overline{\lambda}(\omega)|^2} \end{bmatrix} \tag{3.133}$$

Therefore the problem is reduced to a spectral factorization of the scalar function:

$$f_i(\omega) = 1 + \frac{K^2}{\mid \lambda_i - \overline{\lambda(\omega)} \mid^2} \tag{3.134}$$

$$\text{Let} \qquad R(j\omega;A)\begin{bmatrix} h \\ h' \end{bmatrix} = \begin{bmatrix} \chi \\ \kappa \end{bmatrix} \tag{3.135}$$

with

$$\chi(x,\omega) = \int_0^{\ell} G(\omega, x \mid x')((j\omega - \beta)h(x') - h'(x'))dx'$$

$$\kappa(x,\omega) = j\omega\varphi(x,\omega) - h(x) \tag{3.136}$$

Then

$$CR(j\omega;A)\begin{bmatrix} h \\ h' \end{bmatrix} = k\chi(x,\omega) = K\sum_{n=1}^{\infty}\frac{(j\omega-\beta)h_n - h_n'}{\lambda_n - \lambda(\omega)}\varphi_n(x) \tag{3.137}$$

where h_n and h_n' are the Fourier coefficients of $h(x)$ and $h'(x)$.

Let

$$\underline{w} = G^*C\ R(j\omega;A)\begin{bmatrix} h \\ h' \end{bmatrix} \tag{3.138}$$

$$v_i = \frac{(j\omega-\beta)h_i - h_i'}{\lambda_i - \lambda(\omega)} \tag{3.139}$$

Then

$$w_i = K\frac{\overline{v}_i}{\lambda_i - \overline{\lambda}} = -K^2\frac{(j\omega+\beta)h_i + h_i'}{\mid \lambda_i - \lambda(\omega) \mid^2} \tag{3.140}$$

Define

$$s_i(\omega) = \left[(1 + k^2\tfrac{1}{\lambda_i - \overline{\lambda}(\omega)})^-\right]^{-1} \tag{3.141}$$

Then

$$[F^-(\omega)]^{-1} = \begin{bmatrix} s_1(\omega) & & & 0 \\ & s_2(\omega) & & \\ 0 & & \ddots & \\ & & & s_N(\omega) \end{bmatrix} \tag{3.142}$$

Using the formula for the optimal gains

$$[B^*K]\begin{bmatrix} h \\ h' \end{bmatrix} = \frac{1}{2\pi}\int_{-\infty}^{+\infty} (F^-)^{-1}G^*CR(j\omega;A)\begin{bmatrix} h \\ h' \end{bmatrix} d\omega \tag{3.143}$$

we get

$$[B^*K]\begin{bmatrix} h \\ h' \end{bmatrix} = -\frac{K^2}{2\pi}\begin{bmatrix} \int_{-\infty}^{+\infty} & \frac{s_1(\omega)}{\lambda_1 - \lambda(\omega)|^2}((j\omega+\beta)h_1 + h_1')d\omega \\ & \vdots \\ & \vdots \\ \int_{-\infty}^{+\infty} & \frac{s_N(\omega)}{\lambda_1 - \lambda(\omega)|^2}((j\omega+\beta)h_N + h_N')d\omega \end{bmatrix} \tag{3.144}$$

<u>Theorem 3.4</u>:

consider the multidimensional mode shape control (3.117) for the string.

Define

$$A_i(\omega) = -\int_{-\infty}^{+\infty} \frac{(j\omega+\beta)s_i(\omega)}{\mid \lambda_i - \lambda(\omega) \mid^2} d\omega$$

$$B_i(\omega) = \int_{-\infty}^{+\infty} \frac{s_i(\omega)}{\mid \lambda_i - \lambda(\omega) \mid^2} d\omega \tag{3.145}$$

where $s_i(\omega)$ is given by the expression (3.141). Then the component of the optimal state feedback control is given by:

$$u_i = \frac{K^2}{2\pi}(A_i(\omega)h_i + B_i(\omega)h_i') \tag{3.146}$$

where

$$h_i = \int_0^\ell h(x)\varphi_i(x)dx$$

$$h_i' = \int_0^\ell h'(x)\varphi_i(x)dx \tag{3.147}$$

We therefore see that we have a decoupled control algorithm, $u_i(t)$ dependes on $A_i(\omega), B_i(\omega)$ which depend on $s_i(\omega)$ obtained by performing a scalar spectral factorization.

The real advantage of the approach of this section is the <u>decoupling property</u> In fact we have seen that we have a rational approximation to the transfer function, but this however plays no essential role since we have to deal with scalar spectral factorization only due to the decoupling property and we have already developed a very efficient spectral factorization algorithm based on the Hilbert transform.

To close this section we need to mention the fact that in a physical situation where the controls act on discrete points of the system we need to compute the Fourier coefficients along the eigenbasis. This problem will have a unique solution if the number of controls equals the number of modes we want to control, and a polynomial interpolation will allow the passage from one representation to another.

3.10 Approximations to the Transfer Function

In this section we study the question of "approximations" to the transfer function when the control cannot be decoupled as in the previous section and the Green's function of the system is not available. Of course it is always possible to use numerical integration to approximate the Green's function but this is time consuming and does not yield an efficient algorithm. To prove our point we take somehow the "worst case" i.e. a scalar one point control. In such a situation there is no way of approximating a given shape by a one point except by an averaging process which will not satisfy all boundary conditions.

We prefer not to go into the details of the computations but instead give a summary of the relevant formulae and make some comments.

The problem is:

$$\begin{cases} \dot{\underline{h}} = A\underline{h} + u(t)\begin{bmatrix} 0 \\ \delta(x-\xi) \end{bmatrix} \\ y = Kh \end{cases} \tag{3.148}$$

As we have seen earlier the transfer function is

$$G(j\omega) = -KG(\omega, x \mid \xi) \tag{3.149}$$

and using the bilinaer expression of G we get:

$$G(j\omega) = -K\sum_{n=1}^{\infty} \frac{\varphi_n(x)\varphi_n(\xi)}{\lambda_n - \lambda(\omega)} \tag{3.150}$$

The adjoint operator is:

$$G^*(j\omega)\cdot f = -K\int_0^\ell \overline{G}(\omega, x \mid \xi)f(x)dx \tag{3.151}$$

which has the following expansion

$$G^*(j\omega)\cdot f = -K\sum_{n=1}^{\infty}\frac{\varphi_n(\xi)}{\lambda_n - \overline{\lambda}(\omega)}f_n \tag{3.152}$$

The expression for G^*G is:

$$G^*G(j\omega) = K^2\int_0^\ell \mid G(w, x \mid \xi)\mid^2 dx = K^2\sum_{n=1}^{\infty}\frac{\mid \varphi_n(\xi)\mid^2}{(\lambda_n - \lambda(\omega))(\lambda_n - \overline{\lambda}(\omega))} \tag{3.153}$$

from which the expression of $F(j\omega)$ is deduced.

Again let us mention that the approach given here does not depend on the particular problem of the string, but depends only on the eigenfunctions and eigenvalues of the system.

However a word of caution here; although the series expansion for $F(jw)$, $G^*(jw)\cdots$ are used in an integrated form in the gain formulae (which improves the convergence) there is still a convergence problem to be studied since the bilinear expansion of the Green's function converges in a weak sense only.

3.11 Point Control and Numerical Laplace Transforms

We would like to go back to the problem of point control

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} = a\frac{\partial^2 h}{\partial x^2} + \beta\frac{\partial h}{\partial t} + \gamma h + u(t)\delta(x-\xi) \\ \text{B.C.} \\ \text{I.C.} \end{cases} \tag{3.154}$$

As the Dirac δ-function is the distributional derivative of a function one way to deal with it would be in an integrated form. This way one avoids the convergence problems associated with eigenfunction expansions. Therefore we are going to transform the string equation using Laplace transform and get the solution in the frequency domain by integrating the forcing term against the Green kernel.

Using Laplace transform the equation becomes

$$\alpha\frac{\partial^2 H}{\partial x^2} - (s^2 - \beta s - \gamma)H = (s-\beta)H_0 + H_0' - U(s)\delta(x-\xi)$$
$$H|_{\partial\Omega} = 0 \tag{3.155}$$

The solution is given in terms of the Green's function as

$$H(s,x) = (s-\beta)\int_0^{\ell} H_0(x')G(s,x \mid x')dx' +$$
$$\int_0^{\ell} G(s,x \mid x')H_0'(x')dx' - U(s)G(s,x \mid \xi) \tag{3.156}$$

The displacement is obtained using the inverse Laplace transform:

$$h(t,x) = \frac{1}{2\pi i}\int_{\gamma-i\infty}^{\gamma+i\infty} H(s,x)e^{st}ds \quad Re(s) > \gamma \tag{3.157}$$

where $Re(s) > \gamma$ is the region of convergence.

Therefore

$$h(t,x) = \frac{1}{2\pi i}\Bigg\{\int_{\gamma-i\infty}^{\gamma+i\infty} dse^{st}(s-\beta)\int_0^{\ell} H_0(x')G(s,x \mid x')dx'$$
$$+\int_{\gamma-i\infty}^{\gamma+i\infty} dse^{st}\int_0^{\ell} G(s,x \mid x')H_0'(x')dx'$$
$$-\int_{\gamma-i\infty}^{\gamma+i\infty} U(s)G(s,x \mid \xi)e^{st}ds\Bigg\} \tag{3.158}$$

Let

$$g(t,x \mid x') = \frac{1}{2\pi i}\int_{\gamma-i\infty}^{\gamma+i\infty} G(s,x \mid x')e^{st}ds \tag{3.159}$$

to get

$$\frac{1}{2\pi i}\int_{\gamma-i\infty}^{\gamma+i\infty} (s-\beta)G(s,x \mid x')e^{st}ds = -\beta g(t,x \mid x') - \frac{\partial}{\partial t}g(t,x \mid x') \quad (3.160)$$

Since

$$\frac{1}{2\pi i}\int_{\gamma-i\infty}^{\gamma+i\infty} U(s)G(s,x \mid \xi)e^{st}ds = u(t,x) \quad (3.161)$$

We get the expresssion of $h(t,x)$ in terms of $g(t,x \mid x')$ and $u(t,x)$:

$$h(t,x) = \int_0^{\ell} g(t,x \mid x')(H_0'(x') - \beta H_0(x'))dx'$$
$$- \int_0^{\ell} \frac{\partial}{\partial t}g(t,x \mid x')H_0(x')dx' - u(t,x) \quad (3.162)$$

The first two terms correspond to the initial conditions and the term $u(t,x)$ corresponds to the forcing term.

Now it remains to show how to compute inverse Laplace transforms numerically, although there are algoirthms that do the job directly we prefer to reduce the problem to computation of oscillatory integrals. After some minor modifications the inverse Laplace transform can be written:

$$f(t) = \frac{e^{\gamma t}}{2\pi}\int_{-\infty}^{+\infty} F(\gamma + i\alpha)e^{i\alpha t}\ d\alpha \quad (3.163)$$

$$\text{Let} \qquad F(\gamma + i\alpha) = R(\gamma,\alpha) + i\mathcal{J}(\gamma,\alpha) \quad (3.164)$$

Notice that γ is chosen to insure the convergence of the integral. Plugging the decomposition of $F(\gamma + i\alpha)$ into real and imaginary parts in the expression for $f(t)$ gives:

$$f(t) = \frac{e^{\gamma t}}{2\pi}\int_{-\infty}^{+\infty} [R(\gamma,\alpha)\cos\alpha t - J(\gamma,\alpha)\sin\alpha t]d\alpha \quad (3.165)$$

We have therefore reduced our problem to computations of integrals of the form: $\int_{-\infty}^{+\infty} f(t)\sin\alpha t\ d\alpha$ and $\int_{-\infty}^{+\infty} g(t)\cos\alpha t\ d\alpha$

which can be achieved using existing software.

CHAPTER 4

BEAM EQUATION

In this chapter we move to a more complicated model. Whereas the standard string equation (with no damping) yields simple propagating waves, the theory of beam motion leads to a dispersive system.

In section 4.1 we derive the Euler-Bernoulli equation [4.5] which neglects rotary inertia and shear deflection. We review the propagation of harmonic waves and give the dispersion relation and show the difference with the string equation. From a study of the reflection of waves we get the transmission coefficients. We remark that the end resonance is what is new in the case of the beam and this the mathematical dual of the tunnel effect in Quantum Mechanics (This is similar to the duality of the harmonic oscillator in Quantum Mechanics and the wave guide problem in Electromagnetics). We then study the eigenfunction expansion and give condition for open loop stability for our "beam" (We add to the standard Euler-Bernoulli equation damping and elastic reaciton terms).

In section 4.2 we derive the optimal state feecback control then along a mode shape of the beam.

In section 4.3 we study the question of energy dissipation and show that the presence of damping changes the configuration of the poles of the system with respect to the contour of integration. This study shows the importance of introducing damping to make the residue computations well defined, especially when dealing with operators in an infinite dimensional space. As far as we know there is no infnite dimensional analogue to the Plemelj formulae [26] when some singularities happen to lie on the contour of integration. The introduction of damping also insures the stabilizability of the system (by using semi-group theory as in Chapter 2) and thus avoids the study of the difficult problem of controllability (see [59, 60]). This section is very brief as the literature on this subject is huge and we refer to the references. We do however mention the Chen-Russel [16] model as we use the beam with structural damping in Chapter 5.

In section 4.4 we study the Timoshenko beam model which takes into account rotary inertia and shear deformation and leads to finite phase velocities. This model is excellent agreement with the exact theory based on Elasicity Theory equations, the only way to improve on this model is to introduce nonlinear effects. We give the seires expansion of the solution. We also study the eigenstructure of the associated matrix but we do not try to get an exact formulae for the 4 × 4 transition matrix (as was done for the previous models with 2 × 2 matrices) as this leads to cumbersome computations better left out for a symbolic Language algorithm. This gives us, however, a chance to demonstrate the flexibility of our algorithm by incorporating an ODE software to integrate the differential equations numerically. The results are satisfying (see Fig. [23]). We also give the resolvent equation but stop at this level, just to show the same methods apply and how easily we can handle a variety of models.

4.1 The Euler-Bernoulli Beam

Consider a beam undergoing transverse vibrations with bending moment M and shear force F, satisfying the Euler-Bernoulli assumption: plane cross sections perpendicular to the axis of the beam remain plane and perpendicular to the neutral axis after bending.

The bending moment and the curvature are related as follows:

$$\frac{\partial^2 y}{\partial x^2} = -\frac{M}{EI} \tag{4.1}$$

The bending moment and the shear force are related via

$$F = \frac{\partial M}{\partial x} \tag{4.2}$$

The equation of motion is then

$$\frac{\partial}{\partial x}\left(\frac{\partial M}{\partial x}\right) + f = \rho S \frac{\partial^2 y}{\partial t^2} \tag{4.3}$$

by substitution we get:

$$\frac{\partial^2}{\partial x^2}\left(EI\,\frac{\partial^2 y}{\partial x^2}\right) + \rho\; S\;\frac{\partial^2 y}{\partial t^2} = f(x,t) \tag{4.4}$$

define $\alpha = \frac{EI}{\rho S}$ (constant in the uniform case) to get

$$\frac{\partial^4 y}{\partial x^2} + \frac{1}{\alpha^2}\,\frac{\partial^2 y}{\partial t^2} = f(x,t) \tag{4.5}$$

Besides the fact that we get $\frac{\partial^4}{\partial x^4}$ a partial differential operator of fourth order, the main difference with the string equation is that α does <u>*not*</u> have the dimension of a <u>*velocity*</u>

<u>Remark</u>:

In this case, the translational kinetic energy is

$$dT_y = \frac{1}{2}\,dm\;\times y^2 \tag{4.6}$$

and the rotational kinetic energy is:

$$dT_\theta = \frac{1}{2}\,dJ \times \theta^2 \tag{4.7}$$

To neglect rotary inertia is to omit the contribution of dT_θ to the total energy. This will hold for low frequencies only and will lead to a physical impossibility at high frequencies.

Next we study the <u>propagation of harmonic waves</u> by considering a solution of the form:

$$y(x,\; t) = ae^{i(\gamma x - \omega t)} \tag{4.8}$$

leading to the equation relating frequency and wavenumber:

$$\gamma^4 - \frac{\omega^2}{\alpha^2} = 0 \tag{4.9}$$

In PDE terminology this is the characteristic equation.

There are four roots:

$$\pm\sqrt{\frac{\omega}{\alpha}} \quad \pm i\sqrt{\frac{\omega}{\alpha}} \tag{4.10}$$

The dispersion relation is linear:

$$c = \pm\alpha\gamma \tag{4.11}$$

and give rise to infinite velocities which can be avoided either by considering rotary inertia or by introducing damping. The real roots $\pm\sqrt{\frac{\omega}{\alpha}}$ correspond to simple wave propagation as in the case of the string. The imaginary roots $\pm i\sqrt{\frac{\omega}{\alpha}}$ do not have an equivalent in the case of the string, they are responsible for the dispersion phenomenon and play an important role in the control problem (forced motion).

To get some insight on how waves reflect at boundaries of a beam we consider a pinned end boundary condition:

$$y(0,t) = \frac{\partial^2 y}{\partial x^2}(0,t) = 0 \tag{4.12}$$

and consider the general solution.

$$y(x,t) = A\bar{e}^{i(\gamma x-\omega t)} + Be^{i(\gamma x+\omega t)} + C\bar{e}^{\gamma x}e^{-i\omega t} + De^{\gamma x}\,\bar{e}^{i\omega t} \tag{4.13}$$

with the above boundary condition we get the following transmission coefficients:

$$\frac{B}{A} = \frac{-1+i}{1+i} \qquad \frac{C}{A} = \frac{2i}{1+i} \tag{4.14}$$

Here the C term is what is new in the case of a beam and is referred to as end resonance.. It is suprising to note the duality of this result to that obtained for a particle encountering a *potential well*. In an analoguous way the appearance of the C term is a *quantum* effect called the "tunnel effect".

Next we consider the defining equation for the Green's function:

$$\frac{d^4G}{dx^4} - \beta^4\, G = \frac{1}{EI}\ \delta(x-\xi) \tag{4.15}$$

where $\beta^4 = \frac{\omega^2}{\alpha^2}$ (or more generally: $-\beta^4 = \frac{s^2}{\alpha^2}$)

which leads to two different expressions:

$$\begin{aligned} x<\xi\ \ G_1(x) &= A_1\ sin\ \beta x\ +\ A_2\ cos\ \beta x\ +\ C_1\ sh\ \beta x\ +\ C_2\ ch\ \beta x \\ x>\xi\ \ G_2(x) &= A_1'\ sin\ \beta x\ +\ A_2'\ cos\ \beta x\ +\ C_1'\ sh\ \beta x\ +\ C_2'\ ch\ \beta\ x \end{aligned} \tag{4.16}$$

We have 8 unknowns ($A_1,\ A_2',\ A_2,\ A_2' \ldots c_2'$) and only four boundary conditions:

$$\begin{aligned} y(0) &= \frac{\partial^2 y}{\partial x^2}\,(0) = 0 \\ y(\ell) &= \frac{\partial^2 y}{\partial x^2}(\ell) = 0 \end{aligned} \tag{4.17}$$

We also get three continuity relations:

$$\begin{aligned} G_1\ (\xi-\epsilon) &= G_2(\xi\ +\ \epsilon) && \text{(deflection continuity)} \\ G_1'\ (\xi-\epsilon) &= G_2'(\xi\ +\ \epsilon) && \text{(slope continuity)} \\ G_1''\ (\xi-\epsilon) &= G_2''(\xi\ +\ \epsilon) && \text{(curvature continuity)} \end{aligned} \tag{4.18}$$

The fourth condition expresses the jump of the shear at $x=\xi$:

$$\int_{\xi-\epsilon}^{\xi+\epsilon} y^{(4)}dx - \beta^4 \int_{\xi-\epsilon}^{\xi+\epsilon} y dx = \frac{1}{EI}\int_{\xi-\epsilon}^{\xi+\epsilon} \delta(x-\epsilon)dx = \frac{1}{EI} \tag{4.19}$$

which leads to $y^{(3)}(x)|_{\xi-\epsilon}^{\xi+\epsilon} = \frac{1}{EI}$

The final expression for the Green's function (See [28]) is:

$$\begin{cases} G_1(x/\xi) = A\cdot shx\sqrt{s} & 0\le x\le x' \\ G_2(x/\xi) = B\cdot sh(1-x)\sqrt{s} & \text{otherwise} \end{cases} \tag{4.20}$$

$$\text{where} \qquad A = \frac{-sh(1-x')\sqrt{s}}{s^{3/2}sh\sqrt{s}}$$

$$B = \frac{-sh\sqrt{s}x'}{s^{3/2}sh\sqrt{s}} \tag{4.21}$$

Next we handle the eigenfunction expansion of the beam equation:

$$\begin{cases} \frac{\partial^2 h}{\partial t^2} = a\frac{\partial^4 h}{\partial x^4} + c\frac{\partial h}{\partial t} + dh + u(x,t) \quad a > 0 \quad c < 0 \\ h(0) = h(\ell) = 0 \\ \frac{\partial^2 h}{\partial x^2}(0) = \frac{\partial^2 h}{\partial x^2}(\ell) = 0 \\ \text{I.C.} \end{cases} \tag{4.22}$$

The above boundary conditions correspond to pinned ends, the term $c\frac{\partial h}{\partial t}$ is a viscous damping term having the effect of shifting the poles and leading to finite group velocities (in contrast with Euler-Bernoulli beam). The term dh can be interpreted as the effect induced by an elastic foundation on the beam. Already in the case of a string such term leads to a dispersive system. This term may also lead to mathematical difficulties (non-ellipticity, non-positive definiteness of operators).

As before we introduce a change of dependant variable to "eliminate" the damping term:

$$g(t) = h(t)e^{-\frac{c}{2}t} \tag{4.23}$$

The same transformation holds for the forcing term. We arrive at the following system

$$\begin{cases} \frac{\partial^2 g}{\partial t^2} = a\frac{\partial^4 g}{\partial x^4} + (d + \frac{c^2}{4})g + w \\ g(0) = g(\ell) = 0 \\ \frac{\partial^2 g}{\partial x^2}(0) = \frac{\partial^2 g}{\partial x^2}(\ell) = 0 \\ g_0(x) = h_0(x) \\ g_0'(x) = h_0'(x) - \frac{c}{2}h_0(x) \end{cases} \tag{4.24}$$

The associated eigenvalue problem is:

$$\begin{cases} a\frac{\partial^4 \varphi_n}{\partial x^4} + \delta\varphi_n = \lambda_n \varphi_n \\ \varphi_n(0) = \varphi_n(\ell) = 0 \\ \frac{\partial^2 \varphi_n}{\partial x^2}(0) = \frac{\partial^2 \varphi_n}{\varphi x^2}(\ell) = 0 \end{cases} \tag{4.25}$$

which admits the solution:

$$\varphi_n(x) = \sin\frac{n\pi}{\ell}x \tag{4.26}$$

with eigenvalue

$$\lambda_n = (d + \frac{c^2}{4}) + a(\frac{n\pi}{\ell})^4 \tag{4.27}$$

In the beam case the eigenvalue grow as n^4 which is no surprise since the differential operator is of order 4.

The system will be open loop stable if:

$$\left(d + \frac{c^2}{4}\right) + a\left(\frac{\pi}{\ell}\right)^4 > 0 \tag{4.28}$$

The associated initial value problem is:

$$\begin{cases} a_n''(t) = \lambda_n a_n(t) + w_n(t) \\ a_n(0) = r_n \\ a_n'(0) = v_n \end{cases} \tag{4.29}$$

We refer to chapter [2] for more details.

4.2 Optimal State Feedback Control

In this section we consider the control problem of the system:

$$\begin{cases} \frac{d}{dt}\begin{bmatrix} h \\ h_t \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ a\frac{\partial^4}{\partial x^4} + d & c \end{bmatrix}\begin{bmatrix} h \\ h_t \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \\ h(0) = h(\ell) = 0 \\ h''(0) = h''(x) = 0 \\ h_0(x), h_0'(x) \text{ given} \end{cases} \tag{4.30}$$

which is in the abstract form:

$$\dot{x} = Ax + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \tag{4.31}$$

as a differential equation in a Hilbert space.

We are going to consider optimal state feedback control along modes of the system:

$$u(t) = u_n(t)\varphi_n(x) \tag{4.32}$$

where $\varphi_n(x)$ is an eigenfunction of the system.

The output operator is:

$$Cx = Kh \tag{4.33}$$

We seek to minimize the energy like criterion:

$$J = \int_0^\infty (\|u\|^2 + \|Cx\|^2)dt \tag{4.34}$$

The resolvent equation is:

$$\begin{cases} a\frac{\partial^4 f}{\partial x^4} + [(d+\omega^2) + jc\omega]f = (c - j\omega)\varphi - \Psi \\ g = j\omega f - \varphi \\ f(0) = f(\ell) = 0 \\ f''(0) = f''(\ell) = 0 \end{cases} \tag{4.35}$$

The "modified" transfer function is given by:

$$F(j\omega) = 1 + \frac{K^2}{[\lambda + (d+\omega^2)]^2 + c^2\omega^2} \cdot \int_0^\ell |\varphi_n(x)|^2 dx \tag{4.36}$$

Theorem 4.1:

Consider the mode shape control (4.30) for the "Euler-Bernoulli" beam.

$$\text{Let} \quad h(x,t) = \alpha(t)\varphi_n(x)$$

$$\dot{h}(x,t) = \beta(t)\varphi_n(x) \tag{4.37}$$

Define the integrals.

$$I = \int_{-\infty}^{+\infty} [F^{-}(j\omega)]^{-1} \frac{c - j\omega}{(\lambda + d + \omega^2)^2 + c^2\omega^2} d\omega$$

$$J = \int_{-\infty}^{+\infty} (F^{-}(\omega))^{-1} \frac{d\omega}{(\lambda + d + \omega^2)^2 + c^2\omega^2} \tag{4.38}$$

Then the optimal state feedback control is given by:

$$u(t) = \frac{K^2}{2\pi} \int_{\Omega} |\varphi_n(x)|^2 dx (\alpha(t) I - \beta(t) J) \tag{4.39}$$

4.3 Dissipation of Energy

In the previous sections we considered a simple damping mechanism namely viscous damping which is of the form $c\frac{\partial h}{\partial t}$ (i.e. uniform damping rates). To see how the damping affects the shifting of poles [28] let us consider:

$$\frac{\partial^4 h}{\partial x^4} + c\frac{\partial h}{\partial t} + \frac{1}{\alpha^2}\frac{\partial^2 h}{\partial t^2} = \frac{1}{EI}\delta(x - \xi)e^{-i\omega t} \tag{4.40}$$

Trying a solution of the form, $h(x,t) = h(x) \cdot e^{-i\omega t}$, we get:

$$\frac{d^4 h}{dx^4} - \frac{\omega^2}{\alpha^2} h = \frac{1}{EI}\delta(x - \xi) \tag{4.41}$$

We drop the B.C. for seek of simplicity to be able to solve by Fourier Transform:

$$H(\gamma) = \frac{1}{\sqrt{2\pi}EI}\frac{e^{i\xi\gamma}}{\gamma^4 - \gamma_0^4} \qquad \gamma_0^4 = \frac{\omega^2}{\alpha^2} \tag{4.42}$$

The inverse Fourier transform gives:

$$h(x) = \frac{1}{2\pi EI}\int_{-\infty}^{+\infty} \frac{e^{i\gamma(x-\xi)}}{\gamma^4 - \gamma_0^4} d\gamma \tag{4.43}$$

The integral can be evaluated by residues. However one must take care in choosing a contour of integration. There are four poles: $\gamma = \pm\gamma_0, \pm i\gamma_0$

reals roots: $\pm\gamma_0$ (propagating waves as in string case).

imaginary roots: $\pm i\gamma_0$ (dispersion effect).

For the infinite beam the contour will be as follows:

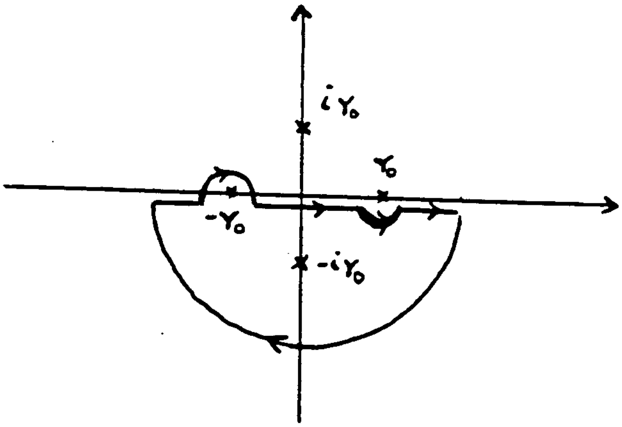


Fig. 21: Pole Configuration for a Beam With No Damping

The pole at γ_0 is being excluded by the radiation condition (i.e. no source at ∞). In the case of the finite beam the B.C. will determine which poles to consider.

What will be the effect of damping?

Consider now the equation with damping:

$$\frac{d^4h}{dx^4} - \left(\frac{\omega^2}{\alpha^2} - ic\omega\right) h = \frac{1}{EI}\delta(x-\xi) \tag{4.44}$$

As before we get

$$h(x) = \frac{1}{2\pi EI}\int_{-\infty}^{+\infty} \frac{e^{-i\gamma(x-\xi)}}{\gamma^4 - \gamma_0^4} d\gamma \tag{4.45}$$

However $\bar{\gamma}_0^2$ is now given by:

$$\bar{\gamma}_0^4 = \frac{w^2}{\alpha^2} - icw \tag{4.46}$$

The pole configuration becomes:

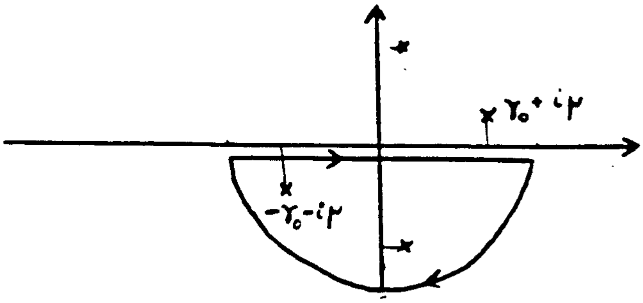


Fig. 22: Pole Configuration for A Beam With Damping

Now the contour of integration includes the desired poles and we need not worry about identations. [28]

With a similar argument one can show that:

$$B^*K = \frac{1}{2\pi}\int_{-\infty}^{+\infty}[F^-(\omega)]^{-1}G^*(j\omega)CR(j\omega, A)d\omega \qquad (4.47)$$

is well defined when we introduce the damping, all the eigenvalues are shifted away from the imaginary axis.

The introduction of damping stems from physical considerations, however it makes the mathematics more complicated but well defined and leads to results in accordance with observation and physical intuition (such as finite responses, finite propagation speeds ...).

In the context of control theory there is another crucial advantage: the damping mechanism acts on the higher order modes, since the controller by its nature acts on a finite number of modes only. However this is not the end of the story, the controller may well exert positive action on higher frequencies which lead to the instability phenomenon called "spillover". As we have explained time and again our control algorithm does not suffer from this drawback. It would be nice however if the damping mechansim (instead of being uniform) were to be stronger as the frequency increases. As a matter of fact experimental studies for the beam have shown that damping is proportional to the frequency (linear relationship). This phenomenon is called structural damping. We will use such model in the sequel when we consider boundary control of beams. Consider a general linear oscillator in a Hilbert space H:

$$\frac{d^2h}{dt^2} + Ah = 0 \tag{4.48}$$

where A is a positive self-adjoint operator on H.

A model for structural damping studied extensively by Chen and Russell [16] is

$$\frac{d^2h}{dt^2} + 2\gamma A^{1/2}\frac{dh}{dt} + Ah = 0 \tag{4.49}$$

In our case $A = \frac{\partial^4}{\partial x^4}$ and hence $A^{1/2} = -\frac{\partial^2 w}{\partial x^2}$ (with paried B.C.). The structural damping term takes the form $-2\gamma\frac{\partial^3 h}{\partial t \partial x^2}$.

When the damping is very high it becomes the dominating term and the equation becomes of diffusion type. This is a singular perturbation changing the type of the system (in addition to the order). This is in complete agreement with numerical experiments we did. We did not pursue the matter further but a computation of the previous integral by residues would show that as ξ increases the solution tends to a Gaussian function which is the diffusion kernel.

4.4 Timoshenko Beam Model

In the case of the Euler-Bernoulli model infinite phase velocities are predicted as well as infinite responses. We mentioned earlier this can be corrected by considering rotary inertia. The Rayleigh theory which takes this effect into account leads to finite phase velocities. Only the Timoshenko model which considers shear effects as well is in excellent agreement with the exact theory.

The Timoshenko beam can be modelled by the coupled system of PDE's:

$$\begin{cases} KGS\left(\frac{\partial \Psi}{\partial x} - \frac{\partial^2 y}{\partial x^2}\right) + \rho S\frac{\partial^2 y}{\partial t^2} = f(x,t) \\ KGS\left(\frac{\partial y}{\partial x} - \Psi\right) + EI\frac{\partial^2 \Psi}{\partial x^2} = \rho I\frac{\partial^2 \Psi}{\partial t^2} \end{cases} \tag{4.50}$$

We can also get the above equations using the variational approach. The potential energy of the beam is:

$$v = \frac{1}{2} \int \left(EI\| \frac{\partial \Psi}{\partial x} |^2 + kGS| \frac{\partial y}{\partial x} - \Psi |^2 \right) dx \qquad (4.51)$$

The kinetic energy is given by:

$$T = \frac{1}{2} \int \left(\rho S \left| \frac{\partial y}{\partial t} \right|^2 + \rho I \left| \frac{\partial \psi}{\partial t} \right|^2 \right) dx \qquad (4.52)$$

By using Hamilton's principle the Timoshenko equations are found to be the Euler-Lagrange equations of the variational problem corresponding to the beam.

The above system represents two coupled modes of deformation:

(i) transverse deflection: $y(x,t)$

(ii) transverse shearing deformation: $\frac{\partial y}{\partial x}(x,t) - \Psi(x,t)$

It can be shown that the system has two families of characteristics, each pair corresponding to two waves of same speed propagating in opposite direction. The above system of coupled wave equations can be transformed into a single equation (under suitable assumptions of differentiability). However it is more convenient to work with the coupled system for the following reasons:

- the expression of the shear force and the bending moment are easier to obtain.

- the boundary conditions can be formulated correctly.

- There are no extra assumptions on differentiability.

The reduced fourth order Timoshenko PDE is:

$$\rho \frac{\partial^2 y}{\partial t^2} - I \frac{\partial^4 y}{\partial t^2 \partial x^2} + EI \frac{\partial^4 y}{\partial x^4} + \frac{\rho}{G} \left(I \frac{\partial^4 y}{\partial t^4} - EI \frac{\partial^4 y}{\partial t^2 \partial x^2} \right) = f(x,t) \qquad (4.53)$$

In this form it becomes clear that the Timoshenko model is a singular perturbation of the Rayleigh model when $\frac{\rho}{G}$ is small. The only way to improve on the Timoshenko model is to introduce a nonlinear system.

To apply our standard techniques. We need to transform the Timoshenko equation into a first order system. Let the equation be:

$$\begin{cases} \alpha\frac{\partial^4\xi}{\partial x^4} + \frac{\partial^2\xi}{\partial t^2} - \beta\frac{\partial^4\xi}{\partial x^2\partial t^2} + \gamma\frac{\partial^4\xi}{\partial t^4} + \nu\frac{\partial\xi}{\partial t} = u(x,t) \\ \text{I.C.} \\ \text{B.C.} \end{cases} \tag{4.54}$$

Define the new variables:

$$\begin{aligned} p(x,t) &= \xi(x,t) \\ q(x,t) &= \dot{\xi}(x,t) \\ r(x,t) &= \ddot{\xi}(x,t) \\ s(x,t) &= \dddot{\xi}(x,t) \end{aligned} \tag{4.55}$$

corresponding to the beam displacement and its first 3 time derivatives. We therefore get the 4th order system:

$$\begin{cases} \frac{d}{dt}\begin{bmatrix} p(x,t) \\ q(x,t) \\ r(x,t) \\ s(x,t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{\alpha}{\gamma}\frac{\partial^4}{\partial x^4} & -\frac{\nu}{\gamma} & -\frac{1}{\gamma} + \frac{\beta}{\gamma}\frac{\partial^2}{\partial x^2} & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \\ s \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u \\ y(x,t) = Kp(x,t) \\ \text{I.C.} \\ \text{B.C} \end{cases} \tag{4.56}$$

As orthogonal basis for the purpose of computation we take:

$$\begin{aligned} \varphi_n(x) &= \sin\frac{n\pi}{\ell}x \\ \lambda_n &= \mu_n^2 \\ \mu_n &= -\left(\frac{n\pi}{\ell}\right)^2 \end{aligned} \tag{4.57}$$

Assuming a series solution of the form:

$$\xi(x,t) = \sum_{n=1}^{\infty} a_n(t)\varphi_n(x) \tag{4.58}$$

We arrive at the initial value problem:

$$\begin{cases} \alpha\mu_n^2 a_n(t) + a_n''(t) - \beta a_n''(t) + \gamma a_n^{(4)}(t) = u_n(t) \\ \text{I.C.} \end{cases} \tag{4.59}$$

In matrix form:

$$\begin{cases} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\beta & 0 & \alpha & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u(t) \\ \text{I.C.} \end{cases} \tag{4.60}$$

there is a slight change of notation when passing from the equation to the system.

Let us define

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\beta & 1 & \alpha & 0 \end{bmatrix} \tag{4.61}$$

The characteristics equation is:

$$\lambda^4 - \alpha\lambda^2 + \beta = 0 \tag{4.62}$$

which is nothing more than an implicit dispersion relation for the Timoshenko wave equation.

Defining:

$$\mu = \lambda^2 \tag{4.63}$$

We get

$$\mu^2 - \alpha\mu + \beta = 0 \tag{4.64}$$

The discriminant of this second order equation is:

$$\Delta = \alpha^2 - 4\beta \tag{4.65}$$

which leads to:

$\underline{\alpha^2 - 4\beta > 0}$:

$$\mu = \frac{+\alpha \pm \sqrt{\alpha^2 - 4\beta}}{2} > 0$$
$$\lambda = \pm\sqrt{\frac{\alpha \pm \sqrt{\alpha^2 - 4\beta}}{2}} \tag{4.66}$$

$\underline{\alpha^2 - 4\beta < 0}$

$$\mu = \frac{\alpha \pm i\sqrt{4\beta - \alpha^2}}{2}$$
$$\lambda = \pm\sqrt{\frac{\alpha \pm i\sqrt{4\beta - \alpha^2}}{2}} \tag{4.67}$$

If we let:

$$\begin{matrix}\sigma \\ \nu\end{matrix} = \frac{\alpha \pm \sqrt{\alpha^2 - 4\beta}}{2} \tag{4.68}$$

then $\lambda = \pm\sqrt{\sigma}, \pm\sqrt{\nu}$

An eigenbasis is given by the 4 vectors.

$$\begin{bmatrix} 1 \\ \sqrt{\sigma} \\ \sigma \\ \sigma\sqrt{\sigma} \end{bmatrix}, \begin{bmatrix} 1 \\ -\sqrt{\sigma} \\ \sigma \\ -\sigma\sqrt{\sigma} \end{bmatrix}, \begin{bmatrix} 1 \\ \sqrt{\nu} \\ \nu \\ \nu\sqrt{\nu} \end{bmatrix}, \begin{bmatrix} 1 \\ -\sqrt{\nu} \\ \nu \\ -\nu\sqrt{\nu} \end{bmatrix} \tag{4.69}$$

We discard the viscous damping in this discussion. The resolvent equation is then given by:

$$\begin{cases} \frac{\alpha}{\gamma}\frac{\partial^4 f_1}{\partial x^4} - \frac{\beta}{\gamma}s^2\frac{\partial^2 f_1}{\partial x^2} - \frac{1}{\gamma}s^2 f_1 = g_4 - \frac{1}{\gamma}(g_1 + g_2) - c\frac{\partial^2}{\partial x^2}(g_1 + g_2) \\ f_2 = sf_1 - g_1 \\ f_3 = sf_2 - g_2 \\ f_4 = sf_3 - g_3 \\ \text{B.C.} \end{cases} \tag{4.70}$$

If we assume the control to be of the form:

$$u(x,t) = u(t)\varphi_n(x) \tag{4.71}$$

scalar control distributed along a shape function of the system. Then we can postulate a solution in the following form:

$$f_j(x,\omega) = \chi_j(\omega)\varphi_n(x) \tag{4.72}$$

Theorem 4.2:

Consider the shape control problem (4.56) for the "Timoshenko"-beam model. Let A be the 4×4 matrix of the associated initial value problem given by (4.61), with implicit dispersion relation:

$$\lambda^4 - \alpha\lambda^2 + \beta = 0 \tag{4.73}$$

Under the assumption $\alpha^2 - 4\beta > 0$ and defining σ, φ as in (4.68), A admits the eigenbasis given in (64). The resolvent equation (4.70) will have the solution (4.72) where:

$$f_1(x,\omega) = \frac{s(t) - (p(t) + q(t))(\frac{1}{\gamma} + \frac{\beta}{\gamma}\lambda_n)}{(\frac{1}{\gamma} + \frac{\beta}{\gamma}\lambda_n) + \frac{2}{\gamma}\lambda_n^2}\varphi_n(x) \tag{4.74}$$

and similar expressions for $f_j(x,\omega)\;\; j = 2,3,4$

Remarks:

1. The "Timoshenko" beam we consider here is more general than that considered in the literature because we add a viscous damping term.
2. We did not get beyond the eigenbasis to commpute the 4×4 transition matrix for two reasons:
   - It is very cumbersome and it is more appropriate for a symbolic language computation (The computations we did in previous chapters involve 2×2 transition matrices which was already a difficult problem).

- As we mentioned before our algorithm is flexible and we wanted to demonstrate it in this example by incorporating an ODE software to solve the beam equations.

3. In the theorem we stop at the resolvent equation, just to show that our algorithm can handle a variety of models (in this case the Timoshenko beam).

As a numerical example to test the software we take a forcing term of the form:

$$u(x,t) = ae^{-t}(\alpha\lambda_n^2 + 1 - \beta + \gamma - \nu)\varphi_n(x) \tag{4.75}$$

generating a displacement function:

$$y(x,t) = ae^{-t}\varphi_n(x) \tag{4.76}$$

where $\varphi_n(x)$ is given by expression (4.57).

As input to the program we need to provide the following information:

- length of the beam: $\ell = 1.5$
- beam parameters: $\alpha = 1.2, \beta = 0.75, \gamma = 1.3, \nu = -1.7$
- absolute and relative errors on computations: $10^{-8}, 0.0$
- time step size and number of periods: $5 \times 10^{-3}, 15$
- assumed shape (eigenfunction): $n = 3$
- desired point x on the beam where the displacement is displayed: $x = 0.95$

The program will then display the exact displacement at x computed from (4.74) and the computed displacement by using (4.58) and integrating (4.59) with the ODE software. The results are reported on the next table and are excellent.

Computed Displacement	Exact Displacement
-0.3074757159	-0.3074757159
-0.3059424162	-0.3059421778
-0.3044174910	-0.3044162691
-0.3029017746	-0.30289779897
-0.3013964891	-0.3013872802
-0.2999031842	-0.2998841107
-0.2984237373	-0.2983884513
-0.2969603837	-0.2969002128
-0.2955156863	-0.2954194248
-0.2940925360	-0.2939460278
-0.2926941514	-0.2924799621
-0.2913241386	-0.2910211980
-0.2899864018	-0.2895697057
-0.2886851132	-0.2881254852
-0.2874248028	-0.2866884470

Fig. 23: Simulation of the Timoshenko Beam

CHAPTER 5

BOUNDARY CONTROL OF BEAM

In this chapter we consider a beam with structural damping, subject to a moment acting on its end (See Fig. 24])

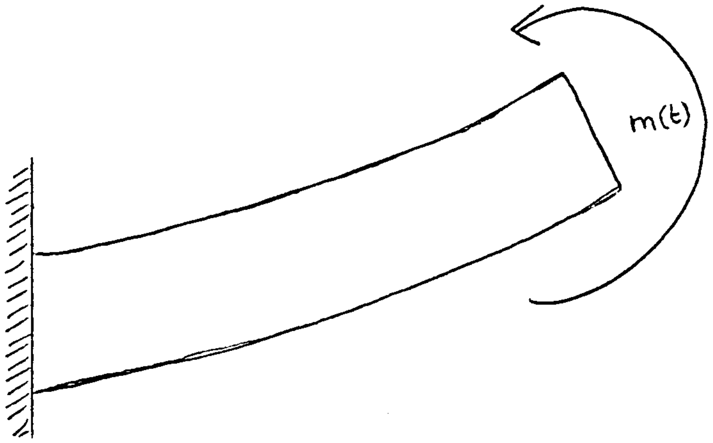


Fig. 24: Beam With Boundary Control

Such beam will satisfy the BVP:

$$\begin{cases} \frac{\partial^2 y}{\partial t^2} = -\alpha \frac{\partial^4 y}{\partial x^4} + \beta \frac{\partial^3 y}{\partial x^2 \partial t} \\ y(0) = y(\ell) = 0 \\ y''(0) = 0, y''(\ell)a = m(t) \qquad \text{B.C.} \\ y_0 \text{ and } \dot{y}_0 \text{ given} \qquad \text{I.C.} \end{cases} \tag{5.1}$$

where $y(x,t)$ is the displacement at x, and $m(t)$ is the moment acting at the end. We want to compute the optial state feedback control $M(t)$ that will minimize the quadratic criterion:

$$\int_0^\infty \left(|m(t)|^2 + k\|y(\cdot,t)\|^2\right) dt \tag{5.2}$$

where $\|y(\cdot,t)\|$ designates the norm of y as a function of x.

In section 5.1 we show the futility of trying to reduce (5.1) to a bounded input map by adding a term of the form:

$$f(x,t) = \varphi(x)u(t) \tag{5.3}$$

to the right hand side of (5.1) for an appropriate function φ. This leads to a different criterion and more importantly because the expression of $u(t)$ involves the derivatives of $m(t)$ and leads to unstable algorithms. Agina the flexibility of our algorithm is clearly demonstrated in this example where we treat this infinite dimensional system with unbounded operators directly and show how "easy" the implementation is.

In section 5.2 we present the method of lines for this problem, just to prove our point of how complicated its implementation is (for instance how to handle the B.C.) with no guarantee of convergence.

In section 5.3 we reduce the PDE (5.1) to a first order symmetric hyperbolic system because these systems have an extensive theory both in the PDE literature (see Courant [17]) and in the Control literature (especially the controllability results by Russel [59]). We again see the inadequacy of yet another method (characteristics) as we arrive at an order 10 system (when actually we have a PDE of order 4) especially when we get in this case only two independent directions.

In section 5.4 we apply our usual technique of transforming the PDE into an infinite system of initial value problems wihtout giving all the details. Next we transform the initial value problem into an equivalent one where the unboundedness of the input operator is moved to the output operator while keeping the same transfer function.

We compute the "modified" transfer function (whose spectral factor enter in the optimal gains fomrulae). Then we compute the optimal gains and give the exact formulae using Theorem 2.13 and we see the huge advantages of this approach:

- No theoretical difficulties such as those encountered with the Riccati Operator equation with unbounded system operators.
- No comparison is even possible from a computational point of view where we have a scalar problem with a closed form formulae when the Riccati equation will lead to huge, stiff systems and no efficient algorithm to solve them (See Chapter 1).

We then implement the system on a minicomputer with the following features:

1. We use a very high gain to demonstrate beyond any doubt the superior performance of our algorithm, our gain was 10,000 which will destabilize most existing algorithms. The free beam does not even have time to react when the controlled beam has been brought to rest (we need to put constraints on the size of the gain though but for physical considerations only not to damage the structure).
2. Again to show the flexibility of the method we use an ODE software to solve the initial value problem (just as with the Timoshenko beam in Chapter 4) even though we are able to solve it exactly.
3. The numerical results demonstrate clearly the robustness of the algorithm. We already know it is robust with respect to parameter variations. Here we test instead its robustness with respect to unmodelled dynamics by taking only 10 modes for the control while keeping 20 modes for the displacement.

Finally we give some insight on the difficulty of boundary control of systems.

5.1 Reduction of the Boundary Control Problem of the Beam Problem to a Bounded Input Operator Control

Let us consider the problem:

$$\begin{cases} \frac{\partial^2 v}{\partial t^2} = -\alpha \frac{\partial^4 v}{\partial x^4} + \beta \frac{\partial^3 v}{\partial x^2 \partial t} + f(x,t) \\ v(0) = v(\ell) = 0 \\ \frac{\partial^2 v}{\partial x^2}(0) = 0 \quad \frac{\partial^2 v}{\partial x^2}(\ell) = 0 \end{cases} \tag{5.4}$$

Note that this problem has homogeneous boundary conditions. If we can write $f(x,t) = \varphi(x)u(t)$ where φ is an L^2-function such that (5.1) and (5.4) are equivalent, then we have succeeded in reducing our original problem to one that we can handle by the usual approach.

Let us take $\varphi(x)$ such that:

$$\varphi(0) = \varphi(\ell) = 0$$

$$\varphi''(0) = 0 \qquad \varphi''(\ell) = 1$$

$$\text{(For instance } \varphi(x) = \frac{x}{6\ell}(x^2 - \ell^2), \varphi''(x) = \frac{x}{\ell}\text{)} \tag{5.5}$$

Then φ satisfies the required conditions.

The computation of the control however suffers from two setbacks:

1. There is no way one can minimize the original quadratic criterion (which involves the square of the moment $|m(t)|^2$), a new criterion is minimized and the relationship between the two is not obvious.

2. The recovery of $m(t)$ from $u(t)$ cannot be done in a stable way since the expression of $u(t)$ involves derivatives $\dot{m}(t)$ and $\ddot{m}(t)$ of $m(t)$.

   (to be convinced of these two claims just assume that $u(x,t) = v(x,t) + m(t)\varphi(x)$ and do computations).

5.2 Method of Lines

If we define $v = \frac{\partial u}{\partial t}$ in (5.1) we get the following system:

$$\begin{cases} \frac{\partial u}{\partial t} = v \\ \frac{\partial v}{\partial t} = -\alpha \frac{\partial^4 u}{\partial x^4} + \beta \frac{\partial^2 v}{\partial x^2} \\ u''(\ell) = m(t) \\ \text{B.C.} \end{cases} \tag{5.6}$$

Defining $h = \Delta x = x_{i+1} - x_i$ we get:

$$\begin{cases} \frac{\partial^2 v}{\partial x^2} = \frac{v_{i+1}(t) - 2v_i(t) + v_{i-1}(t)}{h^2} \\ \frac{\partial^4 u}{\partial x^4} = \frac{u_{i+2}(t) - 4u_{i+1}(t) + 6u_i(t) - 4u_{i-1}(t) + u_{i-2}(t)}{h^4} \\ \text{B.C.} \end{cases} \tag{5.7}$$

The original problem becomes equivalent to:

$$\begin{cases} \dot{u}_i = v_i \\ \dot{v}_i = -\frac{\alpha}{h^2}(v_{i+1} - 2v_i + v_{i-1}) \\ \qquad +\frac{\beta}{h^4}(u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2}) \\ i = 1, 2 \dots N+1 \\ \text{B.C.} \end{cases} \tag{5.8}$$

However we are faced with the following problem:

For $i = 1 (x = 0)$ we need also $u(-h,t), u(-2h,t)$ to compute $\frac{\partial^4 u}{\partial x^4}$; and $v(-h,t)$ to compute $\frac{\partial^2 v}{\partial x^2}$. For $i = N+1 (x = \ell)$ we need $u(\ell+h,t)$ and $u(\ell+2h,t)$ and $v(\ell+h,t)$. One can extend the beam beyond $[0,\ell]$ along the tangents as a scheme to get the necessary boundary conditions.

The implementation is complicated and the convergence of the method is not guaranteed.

5.3 Reduction of the PDE to a 1st Order System

In studying hyperbolic PDE's of order higher than 2 one reduces them to a system of first order PDE's in an attempt to use the method of characteristics to solve the equation.

Following Courant and Hilbert Mathematical Methods for Physics, we define:

$$p^{0,0} = u; p^{0,1} = \frac{\partial u}{\partial x}; p^{0,2} = \frac{\partial^2 \mu}{\partial x^2}; p^{0,3} = \frac{\partial^3 \mu}{\partial x^3}, p^{1,1} = \frac{\partial^2 u}{\partial t \partial x}$$

$$p^{1,2} = \frac{\partial^3 u}{\partial t \partial x^2}; p^{1,0} = \frac{\partial u}{\partial t}, p^{2,0} = \frac{\partial^2 u}{\partial t^2}, p^{3,0} = \frac{\partial^3 u}{\partial t^3}; p^{2,1} = \frac{\partial^3 u}{\partial t^2 \partial x} \tag{5.9}$$

or more generally

$$p^{i,j} = \frac{\partial^{i+j}}{\partial t^i \partial x^j} \tag{5.10}$$

The order of the PDE is $k = 4$, therefore we have $\frac{k(k+1)}{2} = 10$ new variables. And the original PDE is replaced by the new system:

$$\text{For} \qquad i + j = 0, 1, 2, \dots k-2$$

$$\frac{\partial}{\partial t} p^{i,j} = p^{i+1,j} \tag{5.11}$$

$$\text{For} \qquad i+j=k-1 \qquad i \neq k-1$$

$$p_t^{i,j} = p_x^{i+1,j-1}(5.12)$$

Finally the original equation becomes:

$$p_x^{0,3} - \frac{\beta}{\alpha} p_x^{1,1} + \frac{1}{\alpha} p^{2,0} = 0 \tag{5.13}$$

If we introduce the vector notation we get:

$$v(x,t) = (p^{i,j}(x,t)) \tag{5.14}$$

and the system can be written:

$$\frac{\partial U}{\partial t} + A \frac{\partial U}{\partial x} + BU + C = 0 \tag{5.15}$$

which is a symmetric hyperbolic system

$$\text{where} \qquad A = \left[\begin{array}{cccccc} a_1 & a_2 & \dots & a_{k-1} & a_k & \dots \\ -1 & 0 & & & & \\ 0 & -1 & & & & 0 \\ & & & -1 & 0 & \\ 0 & & 0 & 0 & 0 & \dots \end{array} \right] \tag{5.16}$$

The system in 10 × 10, but one can show that an equation of order 4 may be transformed into a 4 × 4 system with a judicious choice of the variables.

The method of characterstics is based on the fact that there are k independent characteristic directions, and one solves the equations:

$$\frac{dx}{dt} = \lambda_i(x,t) \tag{5.17}$$

where λ_i are the eigenvalues of A.

Restricted to the characteristics the PDE becomes a system of ODE's. Unfortunately in our case we have only 2 independent directions because we have one multiple zero eigenvalue:

$$\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = 0 \tag{5.18}$$

5.4 Expansion of the Boundary Optimal Control of a Beam with Structural Damping

$$\begin{cases} \frac{\partial^2 y}{\partial t^2} = -\alpha\frac{\partial^4 y}{\partial x^4} + \beta\frac{\partial^3 y}{\partial x^2 \partial t} \\ y(0) = y(\ell) = 0 \\ y''(0) = 0 \qquad y''(\ell) = u(t) \\ \text{with } Ky \text{ as output} \end{cases} \tag{5.19}$$

where α, β are positive constants.

Let us expand the displacement function into its Fourier series:

$$y(x,t) = \sum_{n=1}^{\infty} y_n(t) \cdot \sin\frac{n\pi}{\ell}x \tag{5.20}$$

Let the initial conditions have the expansion:

$$y_0(x) = \sum_{n=1}^{\infty} r_n\varphi_n(x) \qquad \dot{y}_0(x) = \sum_{n=1}^{\infty} s_n\varphi_n(x) \tag{5.21}$$

where

$$\mu_n = \frac{n\pi}{\ell} \tag{5.22}$$

Thus we arrive at the initial value problem:

$$\begin{cases} \ddot{y}_n(t) = -\alpha\mu_n^4 y_n(t) - \beta\mu_n^2\dot{y}_n(t) + (-1)^n\alpha\mu_n u(t) \\ y_n(0) = r_n \\ \dot{y}_n(0) = s_n \end{cases} \tag{5.23}$$

An equivalent formulation is:

$$\begin{cases} \frac{d}{dt}\begin{bmatrix} z_n(t) \\ \dot{z}_n(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\alpha\mu_n^4 & -\beta\mu_n^2 \end{bmatrix}\begin{bmatrix} z_n(t) \\ \dot{z}_n(t) \end{bmatrix} + (-1)^n\frac{\alpha}{\mu_n^2}\begin{bmatrix} 0 \\ 1 \end{bmatrix}u(t) \\ \begin{bmatrix} z_n \\ \dot{z}_n \end{bmatrix}(0) = \begin{bmatrix} \frac{r_n}{\mu_n^3} \\ \frac{s_n}{\mu_n^3} \end{bmatrix} \\ y_n(t) = K\mu_n^3 z_n(t) \end{cases} \tag{5.24}$$

We write (5.24) as an infinite dimensional system:

$$\frac{d}{dt}\left[\begin{array}{c} z_1 \\ z_2 \\ \vdots \\ \vdots \\ \hline \dot{z}_1 \\ \dot{z}_2 \\ \vdots \\ \vdots \end{array}\right] = \left[\begin{array}{cccccc} & 0 & & & I & \\ -\alpha\mu_1^4 & & 0 & -\beta\mu_1^2 & & 0 \\ & -\alpha\mu_2^4 & & & -\beta\mu_2^2 & \\ & 0 & \ddots & & 0 & \ddots \end{array}\right] \left[\begin{array}{c} z_1 \\ z_2 \\ \vdots \\ \hline \dot{z}_1 \\ \dot{z}_2 \\ \vdots \end{array}\right] + \left[\begin{array}{c} 0 \\ \\ \hline -\frac{\alpha}{\mu_1^2} \\ \frac{\alpha}{\mu_2^2} \\ \vdots \end{array}\right] u(t)$$

$$\left[\begin{array}{c} y_1(t) \\ y_2(t) \\ \vdots \end{array}\right] = K \left[\begin{array}{ccc} \mu_1^3 & & \\ & \mu_2^3 & 0 \\ & 0 & \ddots \end{array}\right] \left[\begin{array}{c} z_1 \\ z_2 \\ \vdots \\ \vdots \\ \hline \dot{z}_1 \\ \vdots \end{array}\right]$$

$$\left[\begin{array}{c} z_1 \\ z_2 \\ \vdots \\ \hline \dot{z}_1 \\ \dot{z}_2 \\ \vdots \end{array}\right](0) = \left[\begin{array}{c} 0 \\ \vdots \\ r_m/\mu_m^3 \\ \hline 0 \\ \vdots \\ 0 \end{array}\right] \qquad (5.25)$$

The <u>transfer function</u> is given by:

$$G(j\omega) = CR(j\omega; A)b = C(j\omega I - A)^{-1}\underline{b} \qquad (5.26)$$

Thus:

$$G^*(j\omega)G(j\omega) = \underline{b}^*(-j\omega I - A^*)^{-1}C^*C(j\omega I - A)^{-1}\underline{b} \qquad (5.27)$$

$$F(j\omega) = 1 + \underline{b}^*(-j\omega I - A^*)^{-1}C^*C(j\omega I - A)^{-1}\underline{b} \qquad (5.28)$$

The <u>gains</u> are given as follows:

$$[B^*\ K]\underline{z}_0 = \frac{1}{2\pi}\int_{-\infty}^{+\infty}[F^-(j\omega)]^{-1}\,\overbrace{G^*(j\omega)CR}^{K}(j\omega; A)\underline{z}_0\,d\omega \qquad (5.29)$$

If we define:

$$\begin{aligned} K(j\omega) &= b^*(-j\omega I - A^*)^{-1}C^*C(j\omega I - A)^{-1} \\ P(j\omega) &= C(j\omega - A)^{-1} \end{aligned} \tag{5.30}$$

We get:

$$K(j\omega) = b^*P^*P \tag{5.31}$$

Then

$$\begin{cases} F(j\omega) = 1 + K(j\omega)\underline{b} \\ [B^*K]\underline{z}_0 = \frac{1}{2\pi}\int_{-\infty}^{+\infty}(F^-(j\omega))^{-1}K(j\omega)d\omega \cdot \underline{z}_0 \\ u(t) = -[B^*K]\underline{z}(t) \end{cases} \tag{5.32}$$

Next we compute the resolvent operator.

Define:

$$\begin{aligned} A &= j\omega I \\ B &= -I \\ C &= \begin{bmatrix} \alpha\mu_1^4 & & \cdots \\ & \alpha\mu_2^4 & \\ 0 & & 0 \end{bmatrix} \\ D &= \begin{bmatrix} j\omega + \beta\mu_1^2 & \cdots & 0 \\ & j\omega + \beta\mu_2^2 & \cdots \\ 0 & & 0 \end{bmatrix} \end{aligned} \tag{5.33}$$

to get:

$$j\omega I - A = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \tag{5.34}$$

$$AC^{-1}D = \begin{bmatrix} \frac{-\omega^2 + j\omega\beta\mu_1^2}{\alpha\mu_1^4} & 0 \\ 0 & \ddots \end{bmatrix} \tag{5.35}$$

Hence:

$$B - AC^{-1}D = \begin{bmatrix} -1 + \frac{\omega^2 - j\omega\beta\mu_1^2}{\alpha\mu_1^4} & 0 \\ 0 & \ddots \end{bmatrix} \tag{5.36}$$

We shall use the inversion lemma in the following:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} -C^{-1}D(B-AC^{-1}D)^{-1} & -A^{-1}B(D-CA^{-1}B)^{-1} \\ (B-AC^{-1}D)^{-1} & (D-CA^{-1}B)^{-1} \end{bmatrix} \tag{5.37}$$

Next we compute the elements of the inverse matrix

$$C' = (B-AC^{-1}D)^{-1} = \begin{bmatrix} \ddots & & 0 \\ & \frac{\alpha\mu_n^4}{(\omega^2-\alpha\mu_n^4)-j\omega\beta\mu_n^2} & \\ 0 & & \ddots \end{bmatrix} \tag{5.38}$$

$$A' = -C^{-1}DC' = \begin{bmatrix} \ddots & & 0 \\ & \frac{j\omega+\beta\mu_n^2}{(\alpha\mu_n^4-\omega^2)+j\omega\beta\mu_n^2} & \\ 0 & & \ddots \end{bmatrix} \tag{5.39}$$

$$CA^{-1}B = \begin{bmatrix} -\frac{\alpha\mu_1^4}{j\omega} & & 0 \\ & -\frac{\alpha\mu_2^4}{j\omega} & \\ 0 & & \ddots \end{bmatrix} \tag{5.40}$$

$$D-CA^{-1}B = \begin{bmatrix} \ddots & & 0 \\ & j\omega+\beta\mu_n^2+\frac{\alpha\mu_n^4}{j\omega} & \\ 0 & & \ddots \end{bmatrix} \tag{5.41}$$

$$D' = (D-CA^{-1}B)^{-1} = \begin{bmatrix} \ddots & & 0 \\ & \frac{j\omega}{(\alpha\mu_n^4-\omega^2)+j\omega\beta\mu_n^2} & \\ 0 & & \ddots \end{bmatrix} \tag{5.42}$$

$$B' = -A^{-1}BD' = +\frac{1}{j\omega}D' = \begin{bmatrix} \ddots & & 0 \\ & \frac{1}{(\alpha\mu_n^4-\omega^2)+j\omega\beta\mu_n^2} & \\ 0 & & \ddots \end{bmatrix} \tag{5.43}$$

Therefore:

$$R(jw;A) = (jwI-A)^{-1} = \begin{bmatrix} \mathcal{D}\left(\frac{j\omega+\beta\mu_n^2}{(\alpha\mu_n^4-\omega^2)+j\omega\beta\mu_n^2}\right) & \mathcal{D}\left(\frac{1}{(\alpha\mu_n^4-\omega^2)+j\omega\beta\mu_n^2}\right) \\ \mathcal{D}\left(\frac{\alpha\mu_n^4}{(\omega^2-\alpha\mu_n^4)-j\omega\beta\mu_n^2}\right) & \mathcal{D}\left(\frac{j\omega}{(\alpha\mu_n^4-\omega^2)+j\omega\beta\mu_n^2}\right) \end{bmatrix} \tag{5.44}$$

where $\mathcal{D}(\lambda_n)$ designates the diagonal operator with λ_n as the element of the diagonal.

The operator $P(j\omega)$ being defined in (5.31) we have:

$$P^*(\omega)P(\omega) = K^2 \begin{bmatrix} \mathcal{D}\left[\frac{\mu_n^6(\omega^2+\beta^2\mu_n^4)}{(\alpha\mu_n^4-\omega^2)^2+\omega^2\beta^2\mu_n^4}\right] & \mathcal{D}\left[\frac{\mu_n^6(\beta\mu_n^2-j\omega)}{(\alpha\mu_n^4-\omega^2)^2+\omega^2\beta^2\mu_n^4}\right] \\ \mathcal{D}\left[\frac{\mu_n^6(j\omega+\beta\mu_n^2)}{(\alpha\mu_n^4-\omega^2)^2+\omega^2\beta^2\mu_n^4}\right] & \mathcal{D}\left[\frac{\mu_n^6}{(\alpha\mu_n^4-\omega^2)^2+\omega^2\beta^2\mu_n^4}\right] \end{bmatrix} \tag{5.45}$$

Since:

$$K = P^*P \tag{5.46}$$

And the "modified" transfer function is given by:

$$F(j\omega) = 1 + K^2 \sum_{n=1}^{\infty} \frac{\alpha^2\mu_n^2}{(\alpha\mu_n^4 - \omega^2)^2 + \omega^2\beta^2\mu_n^4} \tag{5.47}$$

And if we define:

$$\underline{z}(t) = \left[\begin{array}{c} z_1(t) \\ z_2(t) \\ \vdots \\ \hline \dot{z}_1(t) \\ \dot{z}_2(t) \\ \vdots \end{array}\right] \tag{5.48}$$

Then we have:

$$K(j\omega)\underline{z}(t) = K^2 \sum_{n=1}^{\infty} \frac{(-1)^n\alpha\mu_n^4}{(\alpha\mu_n^4 - \omega^2)^2 + \omega^2\beta^2\mu_n^4} \left[(j\omega + \beta\mu_n^2)z_n(t) + \dot{z}_n(t)\right] \tag{5.49}$$

which we plug in the expression for the gains to get:

$$u(t) = -\frac{K^2}{2\pi} \int_{-\infty}^{+\infty} [F^-(j\omega)]^{-1} \sum_{n=1}^{\infty} \frac{(-1)^n\alpha\mu_n^4}{(\alpha\mu_n^4 - \omega^2)^2 + \omega^2\beta^2\mu_n^4}$$

$$\left[(j\omega + \beta\mu_n^2)z_n(t) + \dot{z}_n(t)\right] d\omega \tag{5.50}$$

Define:

$$[F^{-}(\omega)]^{-1} = \mathcal{R}(\omega) + j\mathcal{J}(\omega) \quad \text{to get:}$$

$$Re\left\{\int_{-\infty}^{+\infty} \frac{[F^{-}(\omega)]^{-1}d\omega}{(\alpha\mu_n^4 - \omega^2)^2 + \beta^2\mu_n^4\omega^2}\right\}$$

$$= \int_{-\infty}^{+\infty} \frac{R(\omega)d\omega}{(\alpha\mu_n^4 - \omega^2)^2 + \beta^2\mu_n^4\omega^2} \tag{5.51}$$

Similarly we get:

$$Re\left\{\int_{-\infty}^{+\infty} \frac{j\omega + \beta\mu_n^2)[F^{-}(\omega)]^{-1}}{(\alpha\mu_n^4 - \omega^2)^2 + \beta^2\mu_n^4\omega^2}d\omega\right\} = \beta\mu_n^2 \int_{-\infty}^{+\infty} \frac{\mathcal{R}(\omega)d\omega}{(\alpha\mu_n^4 - \omega^2)^2 + \beta^2\mu_n^4\omega^2}$$

$$- \int_{-\infty}^{+\infty} \frac{\omega\mathcal{J}(\omega)d\omega}{(\alpha\mu_n^4 - \omega^2)^2\beta^2\mu_n^4\omega^2} \tag{5.52}$$

Define the auxiliary gains functions by:

$$G_R^n(\omega) = \int_{-\infty}^{+\infty} \frac{\mathcal{R}(\omega)d\omega}{(\alpha\mu_n^4 - \omega^2)^2 + \beta^2\mu_n^4\omega^2}$$

$$G_J^n(\omega) = \int_{-\infty}^{+\infty} \frac{\omega\mathcal{J}(\omega)d\omega}{(\alpha\mu_n^4 - \omega^2)^2\beta^2\mu_n^4\omega^2} \tag{5.53}$$

we therefore get the expression of the optimal boundary control:

$$u(t) = \frac{-K^2}{2\pi}\sum_{n=1}^{\infty}(-1)^n\alpha\mu_n^4(z_n(t)(\beta\mu_n^2G_R^n(\omega) - G_J^n(\omega)) + \dot{z}_n(t)G_R^n(\omega)) \tag{5.54}$$

Theorem 5.1:

Consider the boundary control problem [5.1] for the beam with structural damping. Let $\mathcal{R}(j\omega)$ and $\mathcal{J}(j\omega)$ be the real and imaginary parts of the spectral factor:

$$[F^{-}(j\omega)]^{-1} = \mathcal{R}(j\omega) + j\mathcal{J}(j\omega) \tag{5.55}$$

and Define the auxiliary gain functions as in (5.52). Then the optimal state feedback boundary control is given by:

$$u(t) = -\frac{K^2\alpha}{2\pi}\sum_{n=1}^{\infty}\left\{G_R^n(\omega)[\dot{z}_n(t) + \beta\mu_n^2z_n(t)] - G_J^n(\omega)z_n(t)\right\}(-1)^n\mu_n^4 \tag{5.56}$$

5.5 Numerical Simulation

We do not give the detailed formulae for the simulation of the forward system and the control bloc as this has been explained extensively for the previous models we treated, the methodology stays the same. This shows the "ease" with which we can handle a variety of systems (membrane, string, Euler-Bernoulli beam, Timoshenko beam, beam with structural damping,...) and a variety of control schemes (Distributed control, Point Control, Multidimensional Point Control, mode shape control, boundary control ...).

Next we describe the numerical results contained in Figure [25]. We need to provide the rpgoram with the following:

- the length of the beam: $\ell = 1.0$
- the parameters of the beam: $\alpha = 1, \beta = 100$
- the time step size T: $T = 10^{-2}$
- amplitude of the initial perturbation: $A = 0.1$
- index of the desired eigenfunction: $n = 1$
- transfer function estimated spectrum: $\Omega_0 = 10,000$
- control gain desired: $K = 10,000$
- absolute and relative errors on computations: $0.1, 0.1$
- desired point on beam: $x = 0.5$

Figure [25] gives the time, the displacement of the free system without control, and the displacement of the beam under the influence of the control. Figure [25] gives the graphic representation of the previous table. As we can see clearly the results are excellent, the free system has not had a chance to react (staying at about 0.09 on the graph during the first hundred time steps) when the controlled system has already been brought to rest (staying at about 0.0 during the same time period). This result ws achieved because of the high gain ($K = 10,000$ and most likely this will not be possible for other algorithms).

Time	Free System	Controlled System
0.01000	0.09951	0.08005
0.02000	0.09805	0.05393
0.03000	0.09076	0.05031
0.04000	0.09495	0.03824
0.05000	0.08892	0.02880
0.06000	0.09968	0.01456
0.07000	0.10002	-0.00158
0.08000	0.10579	0.01595
0.09000	0.09903	0.02085
0.10000	0.10122	-0.01034
0.11000	0.11433	0.01229
0.12000	0.08196	0.00366
0.13000	0.09347	0.00469
0.14000	0.13148	0.00129
0.15000	0.10448	-0.00604
0.16000	0.09458	0.00370
0.17000	0.11242	0.00100
0.18000	0.10823	-0.00448
0.19000	0.10633	-0.00431
0.20000	0.09612	0.01102
0.21000	0.09671	0.02212
0.22000	0.06436	0.00210
0.23000	0.10643	0.00192
0.24000	0.10787	0.00064
0.25000	0.09521	0.00060

Fig. 25a: Simulation of the Beam With Boundary Control

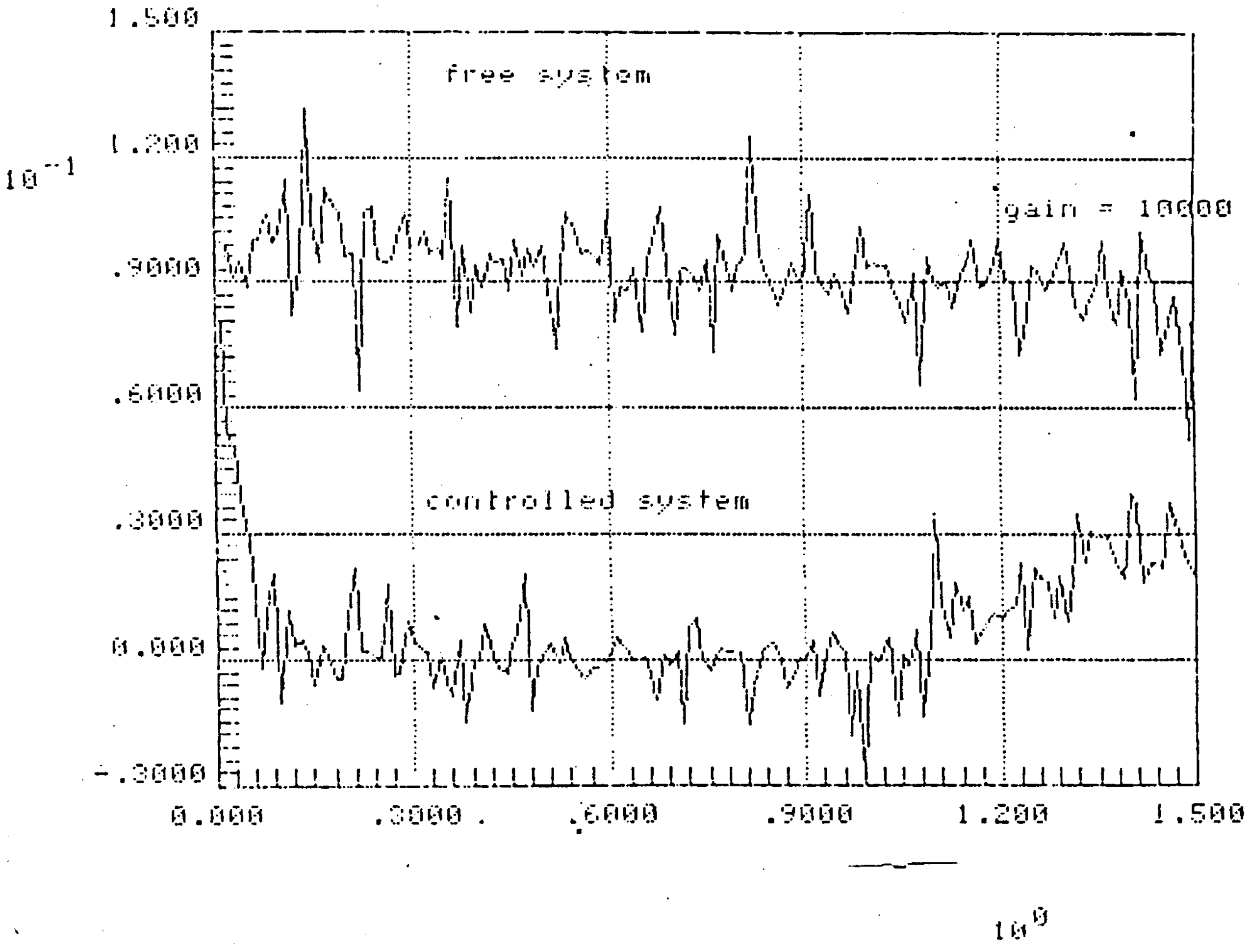


Fig. 25b: Simulation of the Beam With Boundary Control

5.6 Remarks on the Difficulty of the Boundary Control Problem Compared to a δ-Function Control

As we said earlier the difficulty stems from the fact that the input map B is unbounded:

1. in the case of δ-function control, we can write formally

$$Bu(t) = \delta(x - \zeta)u(t) \tag{5.57}$$

and we can approximate δ by L^2-functions, as $\epsilon \to 0 : f_\epsilon(x) \to \delta(x)$ in the weak sense.

Since the fourier coefficients are of the form $n \cos \frac{n\pi}{\ell}\epsilon$, we have to deal with a divergent seires of the form $\sum_n n$ (see previous report) in this case however by using the Green's function one can avoid this problem by working in the frequency domain via Laplace transf. For the string we can write:

$$h(t,x) = G_0(t) + G_0'(t) - \sum_{i=1}^{N} \int_0^\ell G(s, x(\zeta_i)U_i(s)e^{-st}ds \tag{5.58}$$

assuming a control of the forms: $Bu = \sum_{i=1}^N u_i(t)\delta(x - \zeta_i)$

2. In the case of <u>boundary control</u> of the beam we see that we have divergent series of the form $\sum n^2$ (see this report). (It would be as if we were dealing with the derivative of the δ function.)

3. There are serious limitations if we want to study those problems while holding on to the original formulation especially in the case of boundary control of the beam. For instance we will still be faced with the following problem: When B is <u>unbounded</u> what happens to B^*K? Remember that $u(t) = -B^*Kx(t)$. If B^*K is unbounded then there is a problem since even a perfect simulator will not do the job (Any small error on $x(t) \to$ large error on $u(t)$).

CONCLUSIONS

The methodology developed for the computation of optimal state feedback control problem for distributed parameter systems is very powerful and does not suffer any of the drawbacks of heuristic methods (such as spillover, $\cdots$) since it is based on a sound mathematical basis. The main advantages of the method are:

(1) Concise and clear; the steps involved in solving the problem are summarized by the diagram:

(a) Computation of the Green's function

↓

(b) Computation of the transfer function

↓

(c) Solution of the Resolvent equation

↓

(d) Spectral factorization

↓

(e) Optimal Gains

Any of these 5 blocs has subblocs such as supply of eigenvalues, eigenfuncations, $\cdots$.

(2) Very efficient:

(a) The computations of the Gain formulae in the frequency domain by the Wiener-Hopf technique depends only on the number of controllers. By constrast the Riccatti approach depends on the number of modes necessary to approximate the state which is huge in the case of distributed system.

(b) the improvement we introduced by computing the control in a basis of eigenfunctions allowing the decoupling of the control problem (and the forward system as well).

(3) Universality: If we go (1) and investigate the different blocs involved we find that none of the steps depends on the particular system but only on the linearity of the system. The method gives exact formulae in the form of series expansion and allows for numerical computations with arbitrary degree of accuracy. The method has been applied successfully for:

(a) string equation (numerical results)

(b) membrane equation (numerical results)

(c) Euler-Bernoulli beam (numerical results)

(d) Timoshenko-Beam (only necessary formulae)

(e) Beam with structural damping and boundary control. The results were amazingly accurate with the control more and more effective as more and more energy is supplied to the controller.

Another attractive feature of the method is:

(4) Flexibility: it can be applied to more general cases and allows for different approaches to the control problem releiving the designer from the burden of rigid theoretical formulations that can be very hard to realize in practice (such as concatenation of sensors and actuators) which may introduce some potential instabilities in the system.

Futhermore:

(a) When the Green's function of the system cannot be computed (or it is very expensive to do so) the method allows for a scheme that leads to rational approximations to the transfer function.

(b) when the eigenfunctions and eigenvalues are not known the method can allow for the introduction of a finite element scheme to approximate the eigenfunctions of the system without major changes in the methodology.

(c) one can even drop the eigenfunction basis altogether and use finite elements (but decoupling property is lost).

The author also thinks (from his experience with geometric control theory and simulation for nonlinear systems that the method can be generalized to encompass Hybrid Systems (containing both distributed and nonlinear elements).

Finally, we mention that the spectral factorization algorithm we devised could have interesting applications in a variety of Mathematical Physics and Engineering Problems:

(1) Riemann Problem (Hydrodynamics - Elasticity-Fluid dynamics...)

(2) H^∞ design of control systems (minimizing sensitivity functions to get robust controllers).

(3) Singular Integral Equations (Astrophysics...)

(4) Filtering of Distributed Systems

Although a sound theoretical background exists for the above mentioned problems (and many others), efficient numerical algorithms seem to be lacking.

REFERENCES

1. R. A. Adams, "Sobolev Spaces", Academic Press, 1975.

2. A. V. Balakrishnan, "Optimal Control Problems in Banach Spaces," J. Siam Control, Ser. A, Vol. 3, No. 1, USA, 1965.

3. A. V. Balakrishnan, "Compensator Design for Stability Enhancement with Co-located Controllers," presented at the IFAC Conference on the Control of Distributed Parameter Systems, Perpignan, France, June 1989.

4. A. V. Balakrishnan, "Combined Structures-Controls-Integrated Optimization using Distributed Parameter Models," Flight Systems Research Laboratory, UCLA.

5. A.V. Balakrishnan, "A Mathematical Formulation of the SCOLE control problem, Part I," NASA CR 172581, May 1985.

6. M. J. Balas, "Trends in Large Space Structure Control Theory," IEEE Trans, AC-27, No. 3, 1982.

7. M. Balas, "Feedback Control of Flexible Systems," IEEE Tr AC, Vol. AC-23, 1978.

8. H. T. Banks, S.S. Gates, I. G. Rosen, Y. Wang, "The Identification of a Distributed Parameter Model for a Flexible Structure," August 1986, LCDS #86-32, Lefschetz Center for Dynamical Systems.

9. H.T. Banks and I.G. Rosen, "Computational Methods for the identification of spatially varying stiffness and damping in beams," August 1986, LCDS #86-39.

10. John S. Baras, "Frequency Domain Design of Linear Distributed Systems," presented at the 1980 IEEE Decision and Control Conference, Maryland University, Electrical Engineering Dept., 1980.

11. J. S. Baras, and R. W. Brockett, "H^2 Functions and Infinite-Dimensional Realization Theory," SIAMJ Control, 13 (1975), pp. 221-241.

12. J.S. Baras, P. Dewilde "Invariant Subspace Methods in Linear Multivariable-Distributed Systems and Lumped Distributed Network Synthesis," IEEE Proceedings, Vol. 64, No. 1, January 1976.

13. Roger W. Brockett, and Jacques L. Williams, "Operational Methods for Treating Least Squares and Stability Problems in Distributed Parameters Systems," IEEE Conference on Systems, Networks and Computers, Mexico, 1971.

14. A. G. Butkovsky, A.I Egorov and K. A. Lurie, "Optimal Control of Distributed Systems (A Survey of Soviet Publications)," SIAM J. Control, Vol. 6, No. 3, 1968, USA.

15. F. M. Callier, and C. A. Desoer, "An Algebra of Transfer Functions for Distributed Linear Time-Invariant Systems," IEEE Transactions on Circuits and Systems CAS-25(9) pp. 651-662, Sep. 78.

16. G. Chen and D. L. Russell, "A Mathematical Model for Linear Elastic Systems with Structural Damping," Quarterly of Applied Mathematics, January 1982.

17. R. Courant and D. Hilbert, "Methods of Mathematical Physics," John Wiley, 1953.

18. R. F. Curtain, A. J. Pritchard, "Infinite Dimensional Linear Systems Theory," LNCIS8, Springer Verlag, Berlin 1978.

19. R.F. Curtain, "A Synthesis of Time and Frequency Domain Methods for the Control of Infinite Dimensional Systems: A System Theoretic Approach," to appear in the series SIAM Frontiers in Applied Mathematics, 1988.

20. J.H. Davis, B.M. Barry, "A Distributed Model for Stress Control in Multiple Locomotive Trains, Appl. Math. and Optimization," Vol. 3, No. 2/3, pp. 163-190.

21. J.H. Davis, "Hybrid Models for State Estimation in Unit-Train Control," IEEE Tr. Automatic Control, Vol. AC-27, No. 2, April 1982.

22. Jon H. Davis, “Wiener-Hopf Methods for Open-Loop, Unstable Distributed Systems,” SIAM J. Control and Optimization, Vol. 17, No. 6, Nov. 1979.

23. J. H. Davis, and R. G. Dickinson, “Spectral Factorization by Optimal Gain Interaction,” SIAM J. Appl. Math., 43, pp. 289-301, 1983.

24. R. Datko, J. Lagnese, M.P. Polis, “An Example on the effect of time Delays in Boundary Feedback Stabilization of Wave Equations,” SIAM J. Control and Optimization,” Vol. 24, p. 152-156, 1986.

25. H.O. Fattorini, “Second Order Linear Differential Equations in Banach Space,” North Holland, 1985.

26. F.D. Gakov, “Boundary Value Problems,” Dover, 1966.

27. R.P. Gilbert, J.L. Buchanan, “First Order Elliptic Systems: a function theoretic Approach,” Academic Press, 1983.

28. Karl F. Graff, “Wave Motion in Elastic Solids,” Ohio State University Press, 1975.

29. M. J. Grimble, “Extensions to H^∞ Multivariable Robust Controllers and the Relationship to LQG Design Problems,” Int. J. Control, Vol. 50, No. 1, 309-338, 1989.

30. M. J. Grimble, “Optimal H^∞ Robustness and the Relationship to LQG Design Problems. A Solution to the H^∞ Optimal-control Problem is Obtained by Employing Standard LQG Polynomial-based Solution Procedure,” Int. Control, Vol. 43, No. 2, 351-372, 1986.

31. Gilbarg D., Trudinger, “Elliptic Partial Differential Equations of 2nd order,” Springer Verlag, 1977.

32. J. William Helton, “Systems with Infinite Dimensional State Space: The Hilbert Space Approach,” University of California, San Diego.

33. E. Hille, “Analytic Function Theory,” Vol. I, Ginn, Boston, 1959.

34. E. Hille, and R. S. Phillips, "Functional Analysis and Semi-Groups," Colloquium Publications, Vol. 31, American Mathematical Society, Providence RI, 1957.

35. Hochstadt: "Integral Equations", Interscience.

36. Peter C. Hughes, and Robert E. Skelton, "Controllability and Oeservability for Flexible Spacecraft," J. Guidance and Control Vol. 3, No. 5, Article No. 80-4086.

37. F. John, "Partial Differential Equations," Springer Verlag, 1982.

38. Claes Johnson, "Numerical solution of partial differential Equations," Cambridge University Press, 1990.

39. C. Johnson, J. Pitkaranta, "An Analysis of the discontinuous Galerkin method for a scalar hyperbolic equation," Math. Comp., 46, 1986.

40. T. Kailath, "Fredholm Resolvent, Wiener-Hopf Equation and Riccati Differential Equations," IEEE Tr. Inf. Theory, Vol. 15, No. 6, 1969.

41. J. Lagnese, "Boundary Value Controllability of a Class of Hyperbolic Equations in a General Region," SIAM J. Control, 15 (1977), pp. 973-983.

42. Jeffrey H. Lang, and David H. Staelin, "The Computer-Controlled Stabilization of a Noisy Two-Dimensional Hyperbolic System," reprinted from IEEE Transactions on Automatic Control, Vol. AC-27, No. 5, October 1982.

43. Jeffrey H. Lang, and David H. Staelin, "Electrostatically Figured Reflecting Membrance. Antennas for Satellites," IEEE Transactions on automatic Control, Vol. AC 27, No. 3, June 1982.

44. J. H. Lang, "Electrostatically Controlled Wire-Mesh Antenna," Electronics Letters, Vol. 14, No. 20, pp. 665-666, Sept. 1978.

45. Jeffrey H. Lang, "A Perturbation Analysis of Spillover In Closed-Loop Distributed Parameter Systems," IEEE Conference, 1980.

46. P.W. Likins, F.J. Barbera, V. Baddeley, "Mathematical Modeling of Spinning Elastic Bodies for Modal Analysis," AIAA Journal, Vol. 11, No. 9, 1973.

47. V.N. Monakhov, "Boundary Value Problems with Free Boundaries for Elliptic Systems of Equations," American Mathematical Society, No. 57, 1983.

48. L. Meirovitch, "Analytical Methods in Vibrations," Macimillan, New York, 1967.

49. S. G. Mikhlin and S. Prossdorf, "Singular Integral Operators."

50. K.W. Morton, "Generalized Galerkin methods for hypberbolic problems," Comp. Meth. Appl. Mech. Eng., 52, pp. 847-871, 1985.

51. A. Nayfeh, and M. S. Hefzy, "Continuum Modeling of Three-Dimensional Truss-like Space Structures," AIAA Journal, 16, pp. 779-787, 1978.

52. A.K. Noor, M.S. Anderson, W.H. Greene, "Continuum Models for Beam-and Platelike Lattice Structures," Vol. 16, No. 12, Decemberf 1978.

53. P.B. Molinari, "Equivalence Relations for the Algebraic Riccati Equation," SIAM Journal of Control, Vol. 2, No. 2, May 1973.

54. Robert Piché, "Frequency-Domain Continuum Modelling and Control of Third-Generation Spacecraft," University of Waterloo Research Institute, May 1985.

55. Donald A. Pierre, "Minimum Mean-Square-Error Design of Distributed-Parameter Control Systems," ISA Transactions: 5 (263-271).

56. D. A. Pierre, and R. C. Kolb, "Optimal Control of Distributed Parameter Systems," presented at IEEE Region 6 Conference, May 1964.

57. R. Piessen, et al., "Quadpack," Springer Verlag, 1983.

58. J. D. Renton, "The Beam-like Behavior of Space Trusses," AIAA Journal, 22, pp. 273-280, 84.

59. David L. Russell, "Controllability and Stabilizability Theory for Linear Partial Differential Equations: Recent Progress and Open Questions," SIAM Review, Vol. 20, October 78.

60. D.L. Russell, "Nonharmonic Fourier Series in the Control Theory of Distributed parameter Systems, J. Math. Analysis and Appli. 18, pp. 542-560, 1967.

61. Y. Sibuya, "Linear Differential Equations in the Complex Domain: Problems of Analytic Continuation," American Mathematical Society, Vol. 82, 1990.

62. Joe Snoller, "Shockwaves and Reaction-Diffusion Equ.," Springer Verlag, 1983.

63. F.-O. Speck, "General Wiener-Hopf Factorization Methods," Pitman, 1985.

64. Frank Stenger, "The Approximate Solution of Wiener-Hopf Integral Equations," Journal of Mathematical Analysis and Applications 37, 687-724 (1972).

65. R. Temam, "Sur la Stabilite de la Convergence de la Methode des Pas Fractionnaires," Ann. Mat. Pura Appl. LXXXIV, pp. 191-380, 1968.

66. G.P. Tolstov, "Fourier Series," Dover, 1962.

67. F. Treves, "Basic Linear PDE," Academic Press, 1975.

68. R. Triggiani, "On the stabilization problem in Banach Space," J. Math. Analy. Appl., 52 (1975), pp. 383-403.

69. J. C. Willems, "Least Squares Stationary Optimal Control and Algebraic Riccati Equation," IEEE Trans, AC-16, pp. 621-634, 1971.

70. C.J. Weeks, "Shape Determination and Control for Large Space Structures," JPL Publication 81-71, California Institute of Technology, October 1981.

71. Kosaku Yosida, "Some Aspects of E.Hille's Contribution to Semi-group Theory," Integral Equations and Operator Theory, Vol. 4/3, Birkhivser Verlag, Switzerland, 1981.

72. Kosaku Yosida, "Functional Analysis," Springer Verlag, 1980.

73. C. You, E.B. Lee, "Dynamical Boundary Control of Two-Dimensional Wave Equations: Vibrating Membrane on General Domain," IFAC Conference, 1989.

74. E. Zauder, "Partial Differential Equations of Applied Mathematics," J. Wiley, 1983.